AMES GRANT /IN-32

76039

COMPUTER SYSTEMS LABORATORY

STANFORD UNIVERSITY · STANFORD, CA 94305-2192

P.218

CONCURRENT COMMUNICATION AMONG MULTI-TRANSCEIVER STATIONS OVER SHARED MEDIA

Yitzhak Birk

Technical Report: CSL-TR-87-321

March 1987

This report is the author's Ph.D. dissertation which was completed under the advisorship of Professor Fouad A. Tobagi. This work was supported by the National Aeronautics and Space Administration under Grants No. NAG2-292 and No. NAGW-419, the Defense Advanced Research Projects Agency under Contract No. MDA 903-84-K-0249 and an IBM Graduate Fellowship.

CONCURRENT COMMUNICATION AMONG MULTI-TRANSCEIVER STATIONS OVER SHARED MEDIA

Yitzhak Birk

Technical Report: CSL-TR-87-321

March 1987

Computer Systems Laboratory
Departments of Electrical Engineering and Computer Science
Stanford University
Stanford, California 94305-2191

Abstract

Presently, most local-area networks employ a single broadcast bus to interconnect single-transceiver stations. In order to increase a network's throughput beyond a single bus's data rate without using dedicated switching nodes, multiple buses and multi-transceiver stations are required. We explore the design space of single-hop interconnections among such stations; i.e., interconnections that provide a passive transmission path between any two stations. For example, we present interconnections whose throughput can grow ***quadratically*** with the number of transmitters and receivers per station. They consist of a collection of buses, each of which interconnects only a proper subset of the stations using one of their transceivers. Yet, for any two stations, there is at least one bus to which they are both connected. We refer to these as selective-broadcast interconnections, or $\mathcal{SBI}$'s. The use of unidirectional media significantly enriches the design space of $\mathcal{SBI}$'s since, unlike with bidirectional media, the sets of receivers that hear two transmissions need not be identical or disjoint. A graph-theoretic criterion for determining whether or not transmissions over a specified pair of paths would interfere with each other is established. It is then used in studying the performance of various $\mathcal{SBI}$'s. Implementation-related issues, such as power budget in fiber optic implementations, are discussed in the context of local-area networks. Lastly, the concept of $\mathcal{SBI}$'s is shown to also apply to memory-processor interconnection, as well as to additional domains.

A spread-spectrum channel can accommodate several concurrent successful transmissions, and a single-transceiver node can thus utilize only a small fraction of the channel's capacity. In order to allocate the appropriate fraction of capacity to a "busy" node, we propose to equip it with several transmitters and receivers, thereby turning it into a "supernode". Several architectures and operation policies for supernodes are suggested and compared; it is shown that a supernode can significantly outperform a collection of independent conventional nodes with the same total numbers of transmitters and receivers. Packet-radio networks with half-duplex nodes, as well as networks with full-duplex nodes, are considered.

Acknowledgments

I am very grateful to my advisor, Professor Fouad Tobagi, for introducing me to the special problems of high-speed networks and for providing important guidance throughout my research. I have also learned much from him in the way of conveying research results in a clear and enjoyable manner.

I am greatly indebted to Professor Michel Marhic of Northwestern University, who spent a year with our group, for a fruitful and stimulating collaboration during the preliminary stages of the research on selective-broadcast interconnections. Furthermore, Michel's familiarity with optical components and the design constraints that they bring about served as the basis for the research pertaining to the fiber optic implementation of these interconnections.

In carrying out multidisciplinary research, I benefited greatly from my interaction with individuals in various fields. Professor Donald Iglehart, my associate dissertation advisor, introduced me to the theoretical aspects of computer simulation and showed genuine interest in my progress. Professor Jeff Ullman introduced me to computational complexity, suggested issues to be looked into and provided valuable references to related work. Professor Joseph Goodman taught me important optical principles and was always interested and encouraging; he also served on my Ph.D. Orals committee. Professor Michael Flynn was supportive and offered important advice. I also wish to thank Professor G. Leonard Tyler for serving as chairman of my Ph.D. Orals committee, Professor John Cioffi for reading my dissertation, and Professor Adrian Segall of the Technion for getting me started in computer networks.

The financial support provided by a Rothschild Fellowship during my first year and by an IBM Fellowship during the third and fourth years is greatly appreciated. Not being tied to a specific research project, this form of support permitted me to formulate a research topic that truly represented my interests. My research was also supported by NASA (chapters 2,3) and DARPA (chapter 4) through research contracts under the direction of Professor Tobagi.

Finally, I wish to thank members of my family for their parts in this endeavor. My parents, Yehudith and Meir, encouraged me to return to school. My parents in law, Esther and Michael Schaffer, encouraged us to embark on this adventure in spite of the inevitable extended separation. My daughters, Yael and Merav, had to acquire two languages concurrently. Lastly, and most of all, I am grateful to my wife Einat for her unwavering support and continuous love and encouragement.

Contents

Figures

Tables

Chapter 1

Introduction

1.1 Background

Electronic communication networks have evolved over the last several decades [1,2,3,4,5,6]. The first such network was the telegraph network which, due to technological limitations, was only used to carry a limited volume of urgent data traffic. As communication bandwidth increased, real-time voice communication became the primary service, and the telephone network expanded rapidly. In fact, it is still by far the largest in terms of the number of users as well as communication volume.

Since the data sent over the telegraph network was normally created by hand, the telephone service could be viewed as a superior replacement. However, the rapid growth in the computing and data management industries in the last two decades has created a need for efficient means of transmission of "data"; i.e., digitally encoded information that is generated by one computer and intended for another. This has resulted in the rapid evolution of data communication networks, and two major types of such networks have emerged: (i) wide-area networks, (WAN's,) which often span multiple continents but typically interconnect up to several tens of stations, and (ii) local-area networks, (LAN's,) which span up to several kilometers but often interconnect hundreds of stations. The most prominent examples of the two types are ARPANET [7] and Ethernet [8], respectively. Presently, tens of different networks are in existance, and the installed base is many thousands. [9]

Until the early 1980's, data communication networks were separate from the telephone networks. While in some cases leased telephone lines were employed for data transmission, the control was completely separate and the communication channels were not shared. Also, data communication networks were normally owned by their users. More recently, however, these trends have been changing. The desire to reach beyond the boundaries of a single network for purposes such as electronic mail, while at the same time maintaining privacy and security for other applications and preventing network-management complexity from growing, has brought about the concept of internetworking. For example, the nodes of a "public" WAN can serve as gateways to "private" LAN's; a user of one LAN can thus send a message to a user of another LAN via the source LAN, the WAN and the destination LAN. It has also been realized that, while voice traffic and data traffic are different in many ways, the expenditure can often be shared. For example, the cable trenches can be shared.

The separate evolution of data and voice networks was due in part to the fact that voice was handled as an analog signal, whereas computer-generated data is inherently digital. With the gradual transition of the telephone system to digital communication, (voice is sampled and digitized,) this obstacle is being removed. Another difficulty stemmed from the fact that users wanted to own their local-area networks and to tailor them to their own needs; this was impossible with the standard telephone network. However, the recent deregulation of telephone services has resulted in a transition to user-owned local telephone facilities which are connected to the "public" backbone network. This is very similar to the internetworking concept in data networks. Lastly, it has been realized that the two types of traffic actually complement each other in many ways, and the use of a single network for both purposes can thus improve the utilization of network resources. For example, high-volume, low priority data traffic can be deferred to the late night hours, during which the level of voice traffic is very low. The current trend is thus to have a single network that provides integrated services [10,11]. The two primary approaches to achieving this goal are the adaptation of one type of network to also provide the other type of service.

1.2 Characterization of Communication Networks

Communication networks can be characterized by various attributes, such as their function, the protocols used to operate them, their geographical size, the communication media, their topology, etc. For the purposes of this research, the most important attributes are the interconnection network (topology) and the type of communication channels that are employed. The protocols used to operate a network are of lesser importance, and other attributes will not be considered.

1.2.1 Classification of Interconnection Networks

An interconnection network ("interconnection" for short) is the topology of the communication network. It is thus a collection of communication links and, optionally, switches, couplers and buffers. Its purpose is to provide transmission paths among entities which are referred to as stations. Interconnections are required both in communication networks and within computers, particularly multi-processors

[12]. Although the traffic characteristics of these two applications are very different, the basic purpose of the interconnection is the same. Many interconnection schemes have been proposed to date, in both the communication networks and intra-computer environments, and they can be classified as follows:

(i) **Multistage interconnection networks (MIN's) employing dedicated switching nodes.** In such interconnections, the transmitting stations are the inputs, the receiving stations are the outputs, and messages are routed through a succession of intermediate switching nodes whose switches are set according to the destinations of the messages. Examples of such iterconnections are Closs [13], Omega [14], Banyan [15], Benés [16] and the crossbar [13]. With the exception of the crossbar, the number of stages in an MIN is proportional to the logarithm of the number of stations. MIN's were originally used in telephony [13, 16, 17], but in recent years have been adopted by the multi-processor community [12]. They are currently less common in computer networks. Nevertheless, if the telephony approach to providing integrated services prevails, this may change.

(ii) **Interconnections in which the stations themselves also act as forwarding agents.** Here, the stations must be bidirectional. These interconnections are very common in WAN's and in packet[†] radio networks. When used in LAN's, they usually manifest themselves in the form of rings [18], although grids have also been proposed [19]. When used in multiprocessors, they take the form of regular patterns, such as hypercubes [20], cube-connected cycles [21], shuffle-exchange networks [22], multi-dimensional grids and tori [23]. Path lengths in the first three are proportional to the logarithm of the number of stations; in

[†] For various reasons, network protocols often require that messages be chopped into segments of standard length, which are handled independently within the network. These segments are referred to as *packets*.

grids and tori, they are proportional to some root of the number of stations.

(iii) **Single-hop interconnections.** In these, all stations are directly interconnected through passive communication channels, without any intermediate switches; therefore, there is always a path between each pair of stations. Paths need not be disjoint, so not all station pairs can communicate concurrently. Single-hop interconnections are often desirable due to their simplicity, the inherently low latency (no paths need to be established) and the fact that they can be implemented using only passive interconnection components. In some cases, they are inherent to the transmission medium. Two examples of single-hop interconnections are the single broadcast bus, ($\mathcal{SBB}$,) which is used both in local area networks and in computers, and an interconnection consisting of a dedicated link between each pair of stations.

Hybrid interconnections are also possible.

Interconnections can also be classified according to their capability to adapt to the traffic pattern: (i) **Strictly nonblocking** interconnections can accommodate concurrently any collection of source-destination pairs, provided that no source or destination is used more than once; furthermore, the setting of the switches can be determined independently for the different pairs. The crossbar is an example of such an interconnection. (ii) **Rearrangeable nonblocking** interconnections can also accommodate any collection of source-destination pairs, but the setting of the switches for each pair depends on the identities of the other pairs. Examples: Closs and Benés. (iii) **Blocking** interconnections can only accommodate certain collections of source-destination pairs. Examples are Omega and Banyan, as well as all the interconnections in which the stations themselves do forwarding and all single-hop interconnections. (Except for an interconnection consisting of a dedicated link for each pair of stations.)

1.2.2 Characterization of Communication Channels

A channel can serve as a *point-to-point link*, which is used exclusively by one pair of stations; alternatively, it can serve as a *shared medium* which is used by several stations. The network designer may be allowed to choose the type of links, as is often the case in LAN's. In other cases, such as a single-channel network interconnecting mobile radio units, the medium is inherently shared and there is no choice. In practice, point-to-point links are normally used in WAN's, whereas shared media are used in radio networks and in most LAN's.[†]

Along a different dimension, a channel can be used for *narrowband* or *spread-spectrum* communication [24]. With a narrowband channel, the bandwidth occupied by a single transmission is essentially equal to its data rate. Consequently, such a channel can only accommodate a single ongoing transmission; the presence of overlapping transmissions at any given point on the channel constitutes a *collision*, which results in the destruction of all the colliding transmissions. With a spread-spectrum channel, the data stream, of rate B bits per second, is used to modulate a data-independent *code* stream of a much higher rate, $SF \cdot B$. The resultant *chip-stream*, of rate $SF \cdot B$, is used to modulate the carrier of the transmitter. The bandwidth occupied by the transmission is therefore at least SF times higher than the minimum bandwidth required for transmitting data at rate B. (SF stands for spread factor.) At the receiver, which knows the code, the process is reversed and the data is recovered. If used appropriately, the spread-spectrum channel can accommodate several concurrent transmissions. As long as the number of ongoing transmissions is not too high, they are received successfully with high probability.

Spread-spectrum is currently used primarily in military radio networks, due to its favorable anti-jamming and anti-detection properties. (An adversary that does

[†] LAN's are sometimes configured as rings, in which each station has a point-to-point link to each of its two neighbors [18].

not know the code has to jam the entire expanded bandwidth.) A more recent application is in satellite networks, in which spread spectrum permits the use of inexpensive antennas at small ground stations. Such antennas illuminate other satellites as well the one at which they are aimed, but the use of spread spectrum keeps the spectral density of this "pollution" within the permissible limits. Although spread-spectrum is currently not in use in local-area networks, it will probably be used in the future, since channel bandwidth is becoming very inexpensive, yet the cost of the fast electronic circuitry remains prohibitive. With spread-spectrum, the data rate that a single user can sustain need be only a fraction of the aggregate network data rate, and thus only the front end must be fast. A proposal for a fiber-optic implementation of a spread-spectrum channel for LAN's appears in [25].

1.2.3 Network Protocols

A communication network is a shared resource, and this sharing must be governed by some protocol. In the case of WAN's, the functions of the protocol include routing, flow control, and others. In LAN's which use a single channel (bus) to interconnect all stations, the primary function of the protocol is to regulate the access to the bus; such protocols are consequently referred to as *access schemes*. Numerous access schemes have been proposed and analyzed to date [26]. Most of them are distributed protocols, in which each station executes the same algorithm. Differences in the actual behavior of different stations stem from availability or non-availability of data for transmission at each station, from randomization that is sometimes part of the protocol, and from possible differences in the stations' views of the channel.[†] While the first two sources of difference are desirable, the third one is not, since inconsistent views of the channel state may cause two stations

[†] The differences result from the nonzero propagation delay over the channel. A station that is physically close to an event knows about it earlier than a station that is farther away.

to transmit concurrently; this, in turn, may result in a destructive collision. The severity of this problem varies from one access scheme to another; however, for all schemes that make use of the channel status, it increases with an increase in the end-to-end propagation delay as well as with a decrease in the message-transmission time. Similar problems come up in distributed routing protocols in WAN's. There, inconsistency may result in loops or deadlocks.

1.3 Contributions of this Work

1.3.1 Research Thrust

This research focuses on topologies and station architectures for networks that employ *shared media* and provide *single-hop connectivity* among all stations; narrowband as well as spread-spectrum channels are considered. (Most LAN's, radio networks and satellite networks fall into this category.) Presently, most such networks use a single shared communication channel to interconnect all stations, each of which is equipped with a single communication port. (Transmitter and receiver.) All transmissions, regardless of their destination, are heard by all stations.[†] The primary goal of this research is to show that the performance of networks that provide single-hop connectivity via shared media can be enhanced significantly in nonobvious ways if stations are equipped with multiple transmitters and receivers.

[†] In some cases, such as that of single-channel radio networks, a station may be able to hear only a subset of the other stations. As a result, some messages must be forwarded by various stations until they are received by their destinations.

1.3.2 Motivation for Equipping Stations with Multiple Ports

Narrowband channels. With a single narrowband channel, at most one successful transmission can be present at any given point on the bus at any instant. This is true regardless of the access scheme that is employed. Therefore, the transmission rate must at least equal the aggregate network throughput. As a result, very fast channels are often required, resulting in high cost as well as an inability to use certain physical implementations, such as twisted wire-pairs. Also, even the smallest user of the network must be capable of transmitting and receiving at a rate that is equal to the aggregate network throughput. In the common case wherein a LAN interconnects numerous small users, this may cause the cost of the communication interface to dominate the cost of the stations, a clearly unacceptable prospect in many cases. Finally, the efficiency of access schemes drops as the transmission rate increases, and the required transmission rate thus increases faster than the required throughput. In order to obviate these problems, communication must take place concurrently over several buses. To do so while retaining the simplicity of single-hop communication over a passive interconnection network, each station must be equipped with several transmitters and receivers, or ports.

Spread-spectrum channels. As in the case of narrowband channels, the use of multiple channels and, consequently, of multiple ports per station, may be required in order to achieve the desired aggregate throughput. With spread-spectrum channels, however, it may also be necessary to equip certain stations with multiple transmitters and receivers in order to achieve a nonuniform allocation of the channel's capacity to the stations; this may occur even if the channel capacity exceeds the required aggregate throughput. Consider, for example, a single spread-spectrum channel which can accommodate L concurrent transmissions. The channel interconnects a set of stations, and we assume that one of these stations must carry a large fraction of the total traffic. (Such a station might be a gateway, a file server, a mainframe, etc.) Obviously, one would like to allocate to this station a corresponding fraction of the channel capacity. Since a single transmission uses only $\frac{1}{L}$ of the channel's capacity, a single-transceiver station cannot use more than this fraction, and the effective capacity of the network may thus be much lower than the channel capacity. This problem is independent of the details of the spread-spectrum scheme being employed, and can be of utmost importance to the performance of the *network* with such a nonuniform traffic pattern. To permit the station to utilize a larger fraction of the capacity, while using a constant data rate for all transmissions, it must be equipped with multiple transmitters and receivers.

1.3.3 Dissertation Outline

Chapter 2 is devoted to the theoretical study of topologies for single-hop interconnections among stations that are equipped with multiple transmitters and receivers; narrowband channels are assumed. The class of ***selective-broadcast interconnections*** ($\mathcal{SBI}$) is defined to consist of those single-hop interconnections in which each transmission is heard by a proper subset of the stations; several attributes for the characterization of $\mathcal{SBI}$'s are proposed, and various results pertaining to their performance are then derived. For example, it is shown that with a uniform traffic pattern, a large class of $\mathcal{SBI}$'s can accommodate an average number of concurrent transmissions which is proportional to C^2, where C is the number of transmitters and receivers per station. This is a C-fold improvement over the straightforward approach, namely a C-fold replication of the single-bus network. (This replication will be referred to as $\mathcal{PBI}$, for parallel-broadcasts interconnection.) Also, the delay performance of such an $\mathcal{SBI}$ is compared with those of a single bus and of $\mathcal{PBI}$ under various constraints and assuming an ideal access scheme.

The use of unidirectional media is shown to greatly enrich the design space of $\mathcal{SBI}$'s, since the sets of receivers that hear two transmissions may be partly overlapping. (With bidirectional media, they are either identical or disjoint.) A new graph-theoretic criterion is developed for determining whether two concurrent transmissions over a unidirectional interconnection can be received by their respective destinations. This criterion is called *interference*. Mutual noninterference between two paths is a generalization of path-disjointness, the latter being appropriate only to interconnections that employ point-to-point links. Various properties of interference are derived, and the notion of interference is used to derive additional results for unidirectional $\mathcal{SBI}$'s. For example, it is shown that certain such $\mathcal{SBI}$'s can accommodate $N \cdot C/2$ specific transmissions concurrently, (N is the number of stations,) yet they can accommodate an average of C^2 concurrent transmissions with a uniform traffic pattern. The dependence of the performance of certain $\mathcal{SBI}$'s on the traffic pattern is also addressed, as well as fault tolerance. Finally, it is shown that $\mathcal{SBI}$'s are a general concept, which can be applied to various domains in addition to communication networks.

Chapter 3 explores various issues pertaining to the use of $\mathcal{SBI}$'s for communication networks. Implementation as well as operation are considered. Various means of achieving separation between the different subnetworks which constitute an $\mathcal{SBI}$ are considered, and it is shown how to efficiently combine two separation modes. For a fiber optic implementation, which is particularly applicable to high speed networks, it is shown that an $\mathcal{SBI}$ which can accommodate an average of C^2 concurrent transmissions with a uniform traffic pattern is also optimal in terms of power budget. Also, the requirements for passive interconnection components are compared with those of $\mathcal{PBI}$. Finally, the number of stations that can be accommodated (limited by power budget) is compared with those for a single bus and for $\mathcal{PBI}$.

Modifications that are necessary in order to adapt certain $\mathcal{SBI}$'s to operate with certain types of access schemes are discussed. Then, it is shown that the introduction of non-ideal access schemes enhances the relative performance advantage of $\mathcal{SBI}$ over the single bus; the quantification of the advantages of $\mathcal{SBI}$ in chapter 2 is thus shown to be conservative. Finally, the applicability of $\mathcal{SBI}$ to memory-processor interconnections is discussed.

Chapter 4 discusses the architecture and performance of a single station with multiple transmitters and receivers in a spread-spectrum network, focusing on a single-hop topology with a single spread-spectrum channel. Since this station is assumed to be the throughput bottleneck, its throughput determines the network throughput. The emphasis in this research is on the effect of the station's architecture on performance, (rather than on detailed analysis of spread-spectrum channels,) and an attempt is made to extract the issues that have little dependence on the details of the channel. Initially, a slotted-time model is used to study the throughput of this station. It is shown that, while the optimal number of receivers is infinite, the benefit of additional receivers tapers off as the channel capacity becomes the bottleneck. The optimal number of transmitters is finite, since an excessive number of concurrent transmissions by the station will be self-destructive. If a station is half-duplex, (i.e., it cannot receive while any of its transmitters are transmitting,) it is shown that all transmitters should be operated at the same time, or else the station will seldom be available for reception. As an aside, it is also shown that it is sometimes useful to funnel all traffic that is destined to a busy station via two of its neighbors even if the topology provides single-hop connectivity. This can result in a relative throughput increase of up to 36% (compared with direct transmissions) while requiring no additional hardware and only simple, robust protocol modifications.

Next, a more detailed, unslotted model is used in order to determine whether the throughput of an M-receiver station can exceed that of M independent, collocated receivers. Using Markovian analysis, it is initially shown that with a fixed assignment of codes to receivers, the number of different codes that should be assigned to the station in order to maximize the probability of successful reception is a function of the packet-arrival rate. (A receiver can only wait for packets on a single code.) Next, dynamic code-assignment policies are proposed, along with architectures that can support them. They are evaluated using simulation. It is shown that an optimal assignment of codes to the station's M receivers can significantly increase its throughput relative to that of M independent, collocated receivers. Alternatively, fewer transmissions would be required for achieving a given throughput, thus leaving more capacity to other stations.

Chapter 2

Selective-Broadcast Interconnections ($\mathcal{SBI}$'s)

2.1 Introduction

It is often desirable that interconnections be completely passive, and that they provide single-hop connectivity among all stations. Reasons include reliability and minimal latency, as well as simplicity of operation and maintenance. The most prominent such interconnection is a single broadcast bus which interconnects all stations; in fact, this is the only possible single-hop interconnection among stations that are equipped with a single transmitter and receiver. Equipping stations with multiple transmitters and receivers greatly enriches the design space of single-hop interconnections; this chapter is devoted to the exploration of this space.

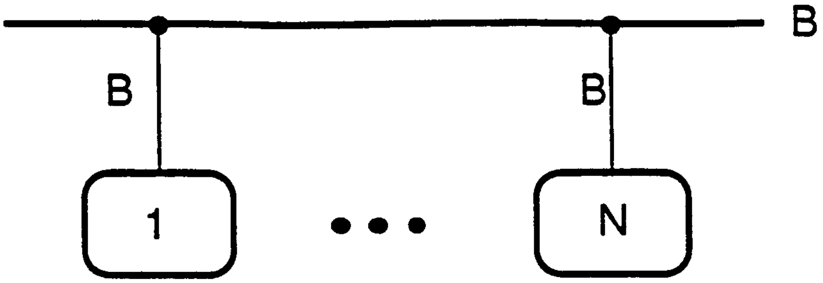


Fig. 2.1 A single broadcast bus with N stations and a transmission rate of B bits per second. ($\mathcal{SBB}$.)

2.1.1 Single-Hop Interconnections

2.1.1.1 The Single Broadcast Bus ($\mathcal{SBB}$)

The most commonly used interconnection for local-area networks is the broadcast bus. Each station has a single transmitter and a single receiver, and they are all connected to the bus; a channel access scheme permits the stations to share the common channel. Fig. 2.1 depicts a single broadcast bus interconnecting N bidirectional stations. The transmission rate and the channel data rate are denoted by B, and the aggregate network throughput (rate of successful transmissions, expressed in bits/sec, summed over all source-destination pairs) is denoted by S. Regardless of the access scheme being employed, the $\mathcal{SBB}$ can be characterized as follows:

- The aggregate network throughput, S, cannot exceed the channel's data rate B. (The channel data rate is limited by the medium.)
- The transmission rate must be at least S; i.e., each transmitter (receiver) must be capable of transmitting (receiving) at a peak rate in excess of S, regardless of the throughput requirements of the station to which it belongs.

- With N stations, the average utilization of transmitters and receivers cannot exceed $1/N$. (Average over stations and time.)
- The power of each transmitter must suffice to feed N receivers. As will be explained in chapter 3, this limits the number of stations that can be accommodated in fiber optic implementations.

From the above inherent characteristics of the $\mathcal{SBB}$, it follows that in order to increase S beyond B, the interconnection must permit some degree of concurrency; i.e., it must accommodate several ongoing transmission. Clearly, the only way of achieving concurrency in single-hop interconnections is through the use of multiple transmitters and/or multiple receivers at each station, along with multiple buses.

2.1.1.2 The Parallel Broadcasts Interconnection ($\mathcal{PBI}$)

A straightforward approach to achieving concurrency is to equip each station with several, say C, transmitters and receivers, and to replicate the $\mathcal{SBB}$ C times. We refer to this as the ***parallel broadcasts interconnection***, or $\mathcal{PBI}$. Fig. 2.2 depicts a $\mathcal{PBI}$ with $C = 3$. In general, a $\mathcal{PBI}$ consists of C subnetworks, each of which interconnects all N stations using one transmitter and one receiver of each station. These subnetworks can be used either independently for bit-serial transmissions, or in parallel for the transmission of C-bit words, (as is the case in address and data buses of computers,) and an access scheme is required to regulate the sharing of the channels [27]. As with $\mathcal{SBB}$, however, there are certain characteristics of $\mathcal{PBI}$ which are independent of the exact access scheme. They are as follows:

- $S \propto C \cdot B$
- The required transmission rate (of an individual transmitter) is only $1/C$ of the aggregate network throughput.
- Average utilization of stations' hardware: $1/N$. (No improvement!)

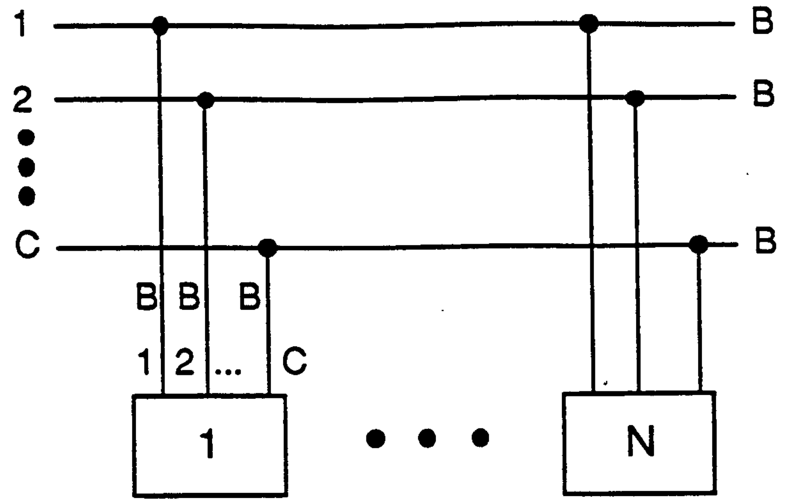


Fig. 2.2 A bidirectional representation of the parallel broadcasts interconnection. ($\mathcal{PBI}$.) C buses, each interconnecting all N stations via one of their transceivers. The transmission rate on each bus is B.

- Power split: N. (No improvement!)

We have thus far established the need for stations with multiple transmitters and receivers whenever the transmission rate of a single transmitter is smaller than the aggregate network throughput. We studied the characteristics of the straightforward approach to achieving concurrency, and saw that the degree of decoupling of the transmission rate from aggregate throughput is proportional to the investment, but there is no improvement in terms of utilization of the stations' hardware or in terms of power budget. This immediately raises the question of whether one can do better. In the remainder of this chapter, it will be shown how one can do much better.

2.1.1.3 Selective-Broadcast Interconnections ($\mathcal{SBI}$'s)

In the straightforward approach, namely $\mathcal{PBI}$ with C buses, there are C different paths between each pair of stations. Since a single path is sufficient in order

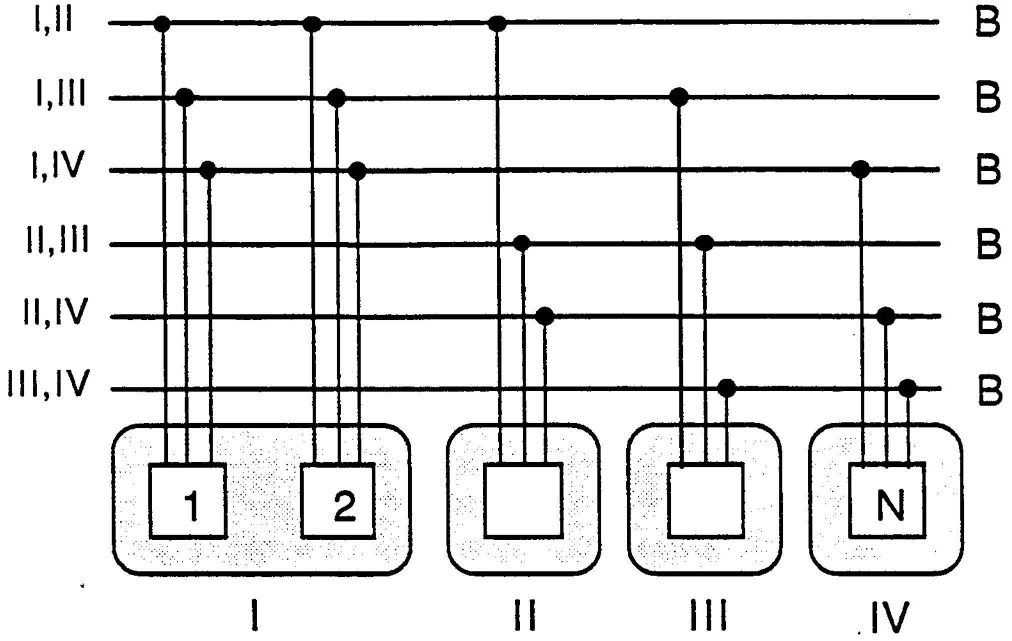


Fig. 2.3 A bidirectional representation of a bidirectional, $(1, C)$-path $\mathcal{SBI}$ with equal-degree, grouping and disjoint subnetworks. $C = 3$. In general, there is a single path between stations that belong to different groups and C paths between any two stations that belong to the same group.

to satisfy the connectivity requirement, this observation suggests that there may be other interconnection topologies which satisfy the connectivity requirement. In such topologies, unlike in $\mathcal{PBI}$, each transmitter is heard by receivers belonging to only a subset of the receiving stations. Whenever a station wishes to transmit a message to some other station, it does so using a transmitter which is heard by some receiver of the recipient as well as by receivers of some other stations; a transmission is thus selectively broadcast to a destination-dependent subset of the stations, and we therefore refer to such topologies as *Selective-Broadcast Interconnections*, or $\mathcal{SBI}$'s.

Fig. 2.3 depicts an $\mathcal{SBI}$ with $C = 3$. This $\mathcal{SBI}$ will later be classified as a $(1, C)$-path, equal-degree, bidirectional $\mathcal{SBI}$. To construct it. the N stations were partitioned into 4 groups, and 6 subnetworks were constructed, each of which interconnects all stations of some two groups. In general, such an $\mathcal{SBI}$ consists of $0.5 \cdot C(C+1)$ subnetworks, thus permitting throughput to increase quadratically with C. This $\mathcal{SBI}$ provides C alternate paths among stations that belong to the same group, but only a single path between any two stations that belong to different groups. We will return to this $\mathcal{SBI}$ in a later section.

Fig. 2.4 depicts a different $\mathcal{SBI}$ with $C = 3$. This $\mathcal{SBI}$ will be classified as a single-path, equal degree unidirectional $\mathcal{SBI}$ with disjoint subnetworks. Here, the stations are partitioned into C groups, and each station is split into the transmit part and the receive part. For each pair of groups, say (i, j), a dedicated subnetwork carries transmissions from stations in i to stations in j. All stations in i and all stations in j are connected to this subnetwork using one of their transmitters and one of their receivers, respectively. This $\mathcal{SBI}$ provides a single path from each station to each other station. Since there are C groups, there are C^2 disjoint subnetworks, and the number of concurrent transmissions that can be accommodated is roughly twice as high as in the bidirectional $\mathcal{SBI}$. This $\mathcal{SBI}$ will be studied in great detail later.

2.1.2 Unidirectional Broadcast Media

A unidirectional broadcast medium is one in which a signal propagates in one direction. This is meaningful only when the medium is unidimensional, as is the case with cables and with optical fibers. In later sections, we will show that unidirectional media enrich the design space of interconnections; also, interesting implementation environments , such as optical fibers, are unidirectional. Lastly, bidirectional media can always be described in terms of unidirectional media, whereas the converse is not true. Therefore, unidirectional media will be assumed from now on.

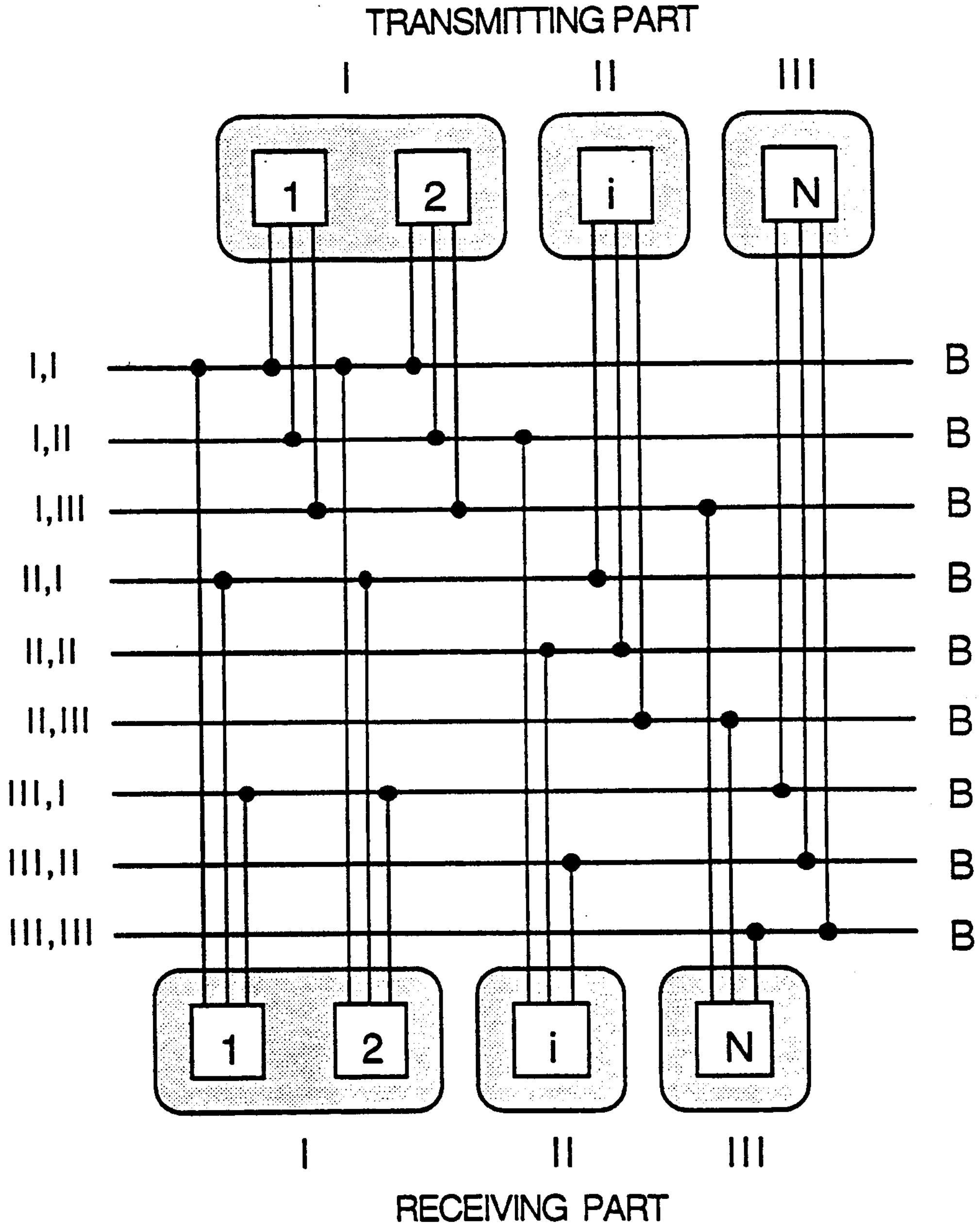


Fig. 2.4 A bidirectional representation of a unidirectional, single path $\mathcal{SBI}$ with equal-degree, disjoint subnetworks and grouping. $C = 3$.

An interconnection that uses unidirectional media can be described as a directed bipartite graph in which nodes on the left represent transmitters, nodes on the right

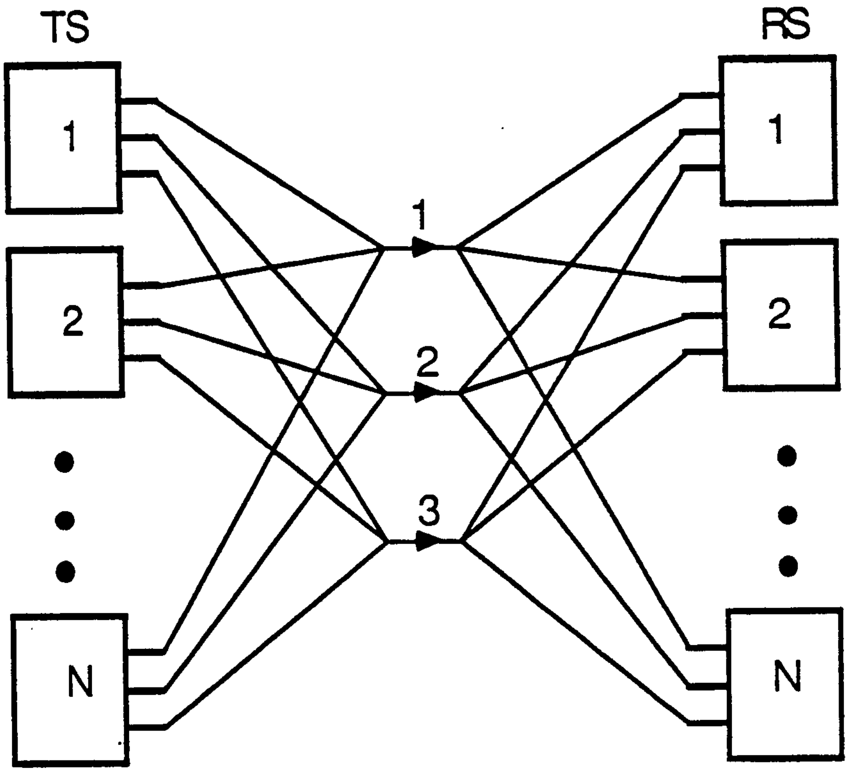


Fig. 2.5 Unidirectional representation of $\mathcal{PBI}$ with $C = 3$. Each box on the left represents the transmitting part of a station; each box on the right represents the receiving half. Lines that are incident on boxes represent individual transmitters and receivers. Intermediate nodes represent passive, directional star couplers.

represent receivers, and there is an edge from node i to node j if and only if receiver j can hear transmissions of transmitter i. The number of edges in the graph can often be reduced sharply if intermediate nodes are added. We refer to those as *directional star couplers*. A signal that enters a coupler through one of its inbound edges exits it over all outbound edges. (A coupler is not a selector.) Figures 2.5 and 2.6 depict unidirectional representations of $\mathcal{PBI}$ with $C = 3$ and of the unidirectional $\mathcal{SBI}$ example, respectively. Each box on the left represents the transmit part of a station, and each box on the right represents the receive part. Edges that are incident on

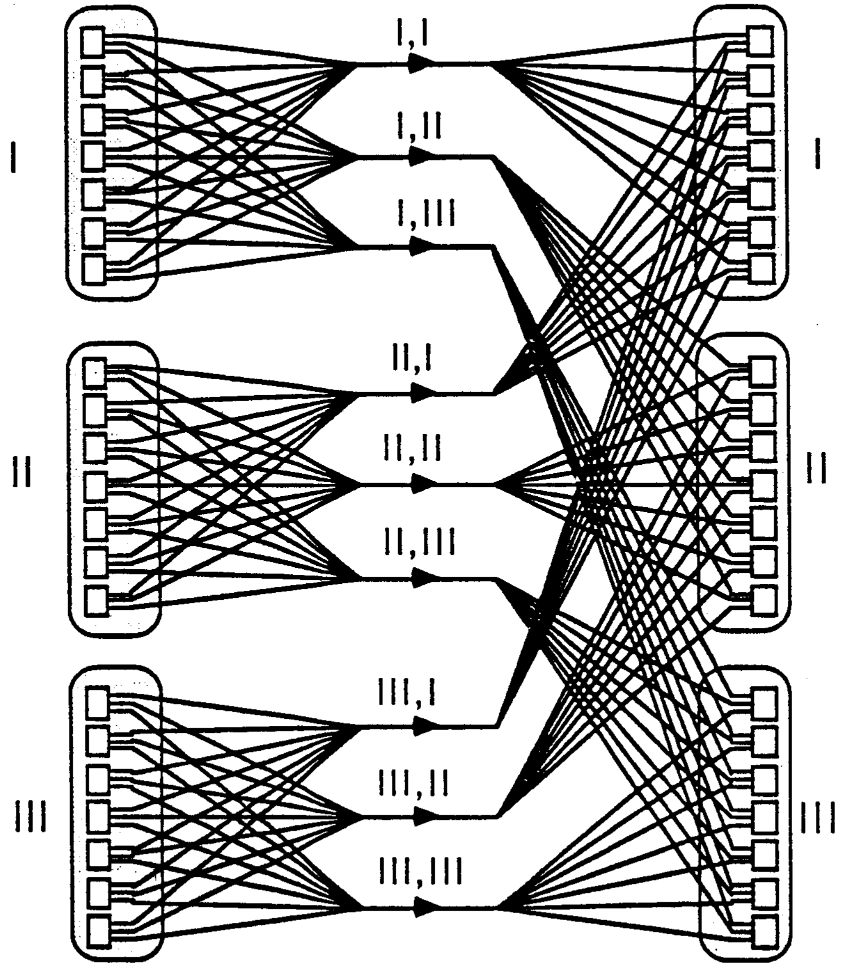


Fig. 2.6 Unidirectional representation of the single-path, unidirectional equal-degree $\mathcal{SBI}$ with disjoint subnetworks and grouping. ($C = 3$.)

boxes represent individual transmitters and receivers.

Since transmitters and receivers are separate in a unidirectional description of an interconnection, it is natural to address the slightly more general problem of interconnecting N_T transmitting stations, (TS's for short,) with N_R receiving stations, (RS's,) so as to permit the transmitting stations to send messages to the receiving stations. In the sequel, we will consider this more general problem, which includes the interconnection of N bidirectional stations as a special case.

2.1.3 Outline of the Remainder of the Chapter

Section 2 defines an $\mathcal{SBI}$, presents a set of attributes which are useful in the characterization and classification of $\mathcal{SBI}$'s, and states the focus of this research. Sections 3 and 4 contain detailed studies of two classes of $\mathcal{SBI}$'s, both of which can be implemented with unidirectional as well as bidirectional media. In order to facilitate the understanding of more general $\mathcal{SBI}$'s, section 5 presents a criterion for concurrency in unidirectional media, and shows why there are interconnections that cannot be implemented with bidirectional media. Section 6 then presents a detailed study of one such class of $\mathcal{SBI}$'s. Section 7 presents various ways of accommodating nonuniform traffic patterns. Section 8 compares the delay performance of a specific class of $\mathcal{SBI}$'s with those of $\mathcal{SBB}$ and of $\mathcal{PBI}$, and section 9 summarizes the chapter.

2.2 $\mathcal{SBI}$ Design Space and Performance Measures

2.2.1 Definition

Given a set of transmitting stations and a set of receiving stations, an $\mathcal{SBI}$ is a single-hop interconnection which provides at least one path from each TS to each RS. Furthermore, each transmission is heard by a proper subset of the RS. Although this definition is very broad and is almost identical to that of a single-hop interconnection, our focus will be on certain classes of $\mathcal{SBI}$'s which have a regular structure and in which the "selective broadcast" feature is very clear. The term $\mathcal{SBI}$ will be used primarily in reference to those topologies, and their performance will be compared with that of $\mathcal{PBI}$.

2.2.2 Classification of $\mathcal{SBI}$'s

An $\mathcal{SBI}$ can be characterized as having or not having various attributes. Some useful ones are:

- All TS's have an equal number of transmitters, and all RS have an equal number of receivers. (Standard stations.)
- The number of disjoint paths between each TS and RS is k. (A k-path $\mathcal{SBI}$.)
- All transmitters reach an equal number of receivers, and all receivers can be reached by an equal number of transmitters. In the bipartite graph representing such an $\mathcal{SBI}$, all nodes on the left have an equal degree, as do all the nodes on the right. (Equal degree.)
- The sets of receivers reached by any two transmitters are either identical or disjoint. (Disjoint subnetworks.)
- Two TS's, say i and j, are said to belong to the same *group* if and only if they have equal numbers of transmitters and, for each transmitter of i, there is a transmitter of j such that the two transmitters reach identical sets of receivers. Similarly for RS's with receivers and transmitters exchanged. (Grouping in the weak sense of TS's and RS's, respectively.)
- If, in addition, transmitters of TS's that belong to different groups reach disjoint sets of receivers, (similarly for RS's,) the grouping is said to be in the strong sense. Since any $\mathcal{SBI}$ can be viewed as consisting of groups of size ≥ 1 "grouping" will generally be used to refer to the strong sense, and an $\mathcal{SBI}$ with grouping is one that has grouping of TS's as well as of RS's. (Grouping in the weak sense will be useful when the groups are of equal sizes.) Note that grouping in the strong sense implies disjoint subnetworks.
- Station i can reach station j over a given subnetwork if and only if j can reach i over the same subnetwork. (Bidirectional $\mathcal{SBI}$.) Note that a bidirectional $\mathcal{SBI}$

always has disjoint subnetworks.

It should be noted that there are $\mathcal{SBI}$'s for which the above attributes are not suitable. For example, an $\mathcal{SBI}$ that provides a single path between one pair of stations but two paths between some other pair cannot be classified as a k-path $\mathcal{SBI}$ for any single k. Nevertheless, the above attributes will prove useful in the sequel. Following are some examples of $\mathcal{SBI}$'s along with their classification. The $\mathcal{SBI}$ in Fig. 2.3 is classified as: standard stations, equal degree, grouping, disjoint subnetworks and bidirectional. As for the number of paths between stations, it can be described (with some abuse of notation) as a "$(1,3)$" $\mathcal{SBI}$, since it provides a single path between stations in different groups and 3 paths between stations in the same group. The $\mathcal{SBI}$'s depicted in Fig. 2.6 and 2.7 are both classified as standard stations, single path, equal degree, disjoint subnetworks and unidirectional. (Unidirectional = not bidirectional.) However, the one in Fig. 2.6 has grouping of TS's as well as RS's, whereas the $\mathcal{SBI}$ in Fig. 2.7 only has grouping of TS's. The $\mathcal{SBI}$ depicted in Fig. 2.8 is unidirectional, single-path, equal degree but without grouping and with overlapping subnetworks. Finally, the $\mathcal{SBI}$ in Fig. 2.9 interconnects nonstandard stations, provides a single path between any two stations, has unequal degree and no grouping or disjoint subnetworks. In the sequel, all $\mathcal{SBI}$'s will have standard stations, unless stated otherwise.

2.2.3 Possible Design Goals

Using stations with multiple transmitters and receivers leaves great flexibility to the designer, which can be used to achieve various goals. For example:

- Given the number of TS, (N_T,) the number of RS, (N_R,) the ($N_T \times N_R$) traffic pattern matrix, the transmission rate, the total number of transmitters and the total number of receivers, design the $\mathcal{SBI}$ that maximizes throughput. Alternatively, given the absolute traffic requirements but not the transmission rate,

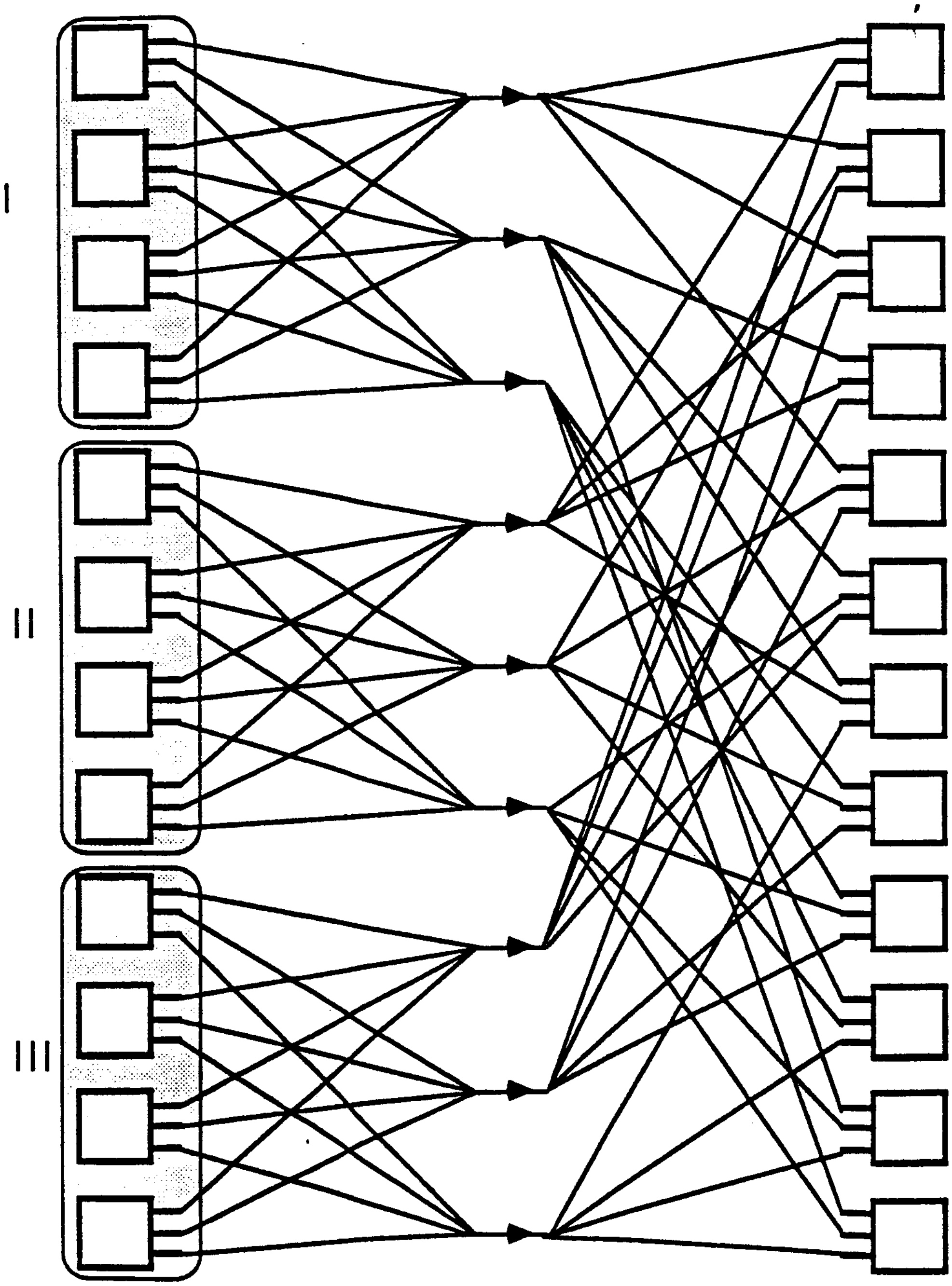


Fig. 2.7 Unidirectional, single path, equal-degree $\mathcal{SBI}$ with disjoint subnetworks and grouping of TS's but without grouping of RS's. $N = 12$; $C = 3$.

design the $\mathcal{SBI}$ which minimizes the transmission rate required to achieve the traffic requirement. This could be the case for a fiber-optic interconnection with

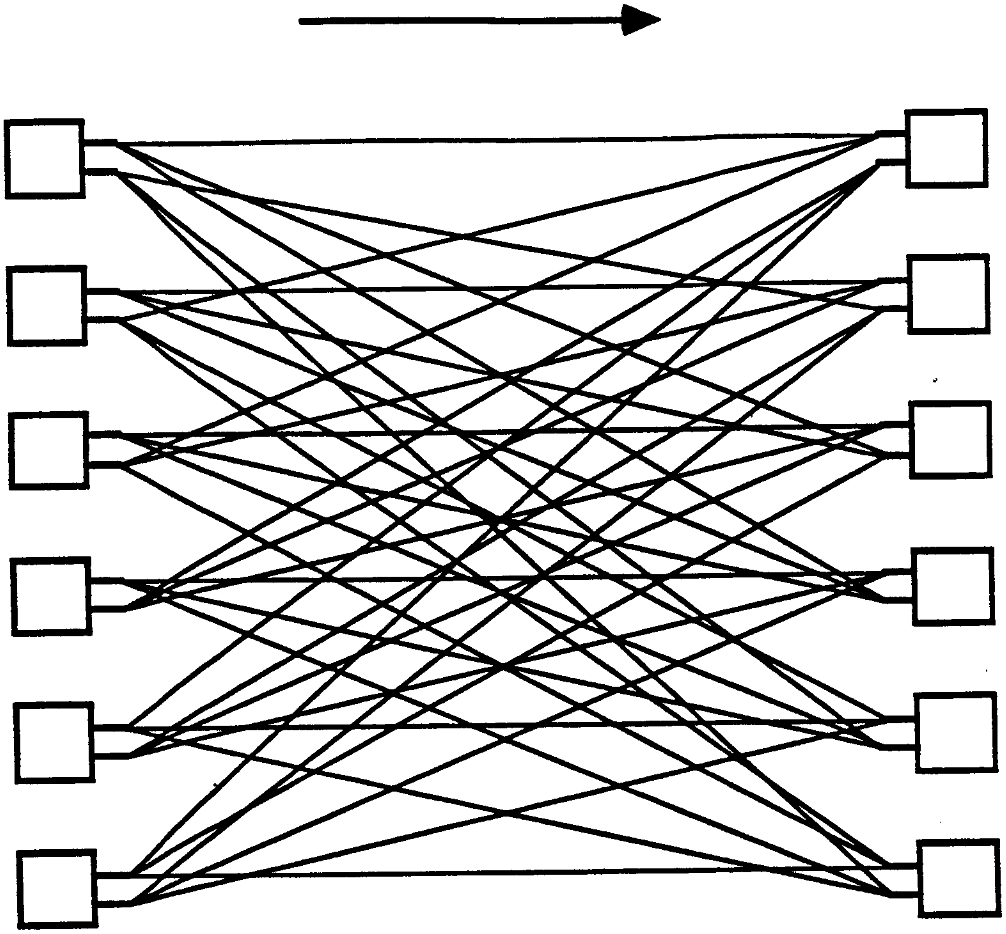


Fig. 2.8 Single path, equal degree $\mathcal{SBI}$. (No grouping, overlapping subnetworks.) $N = 6$; $C = 2$.

an unlimited number of fibers and connectors but a limited number of transmitters and receivers.

- Similar to the above, but optimize the design for some other performance function, such as delay for a given throughput. In some cases, the exact traffic pattern is not known and a performance measure can be defined over a range of patterns. In other cases, the cost of interconnection components is of major concern. In some applications, it makes sense to permit different transmission

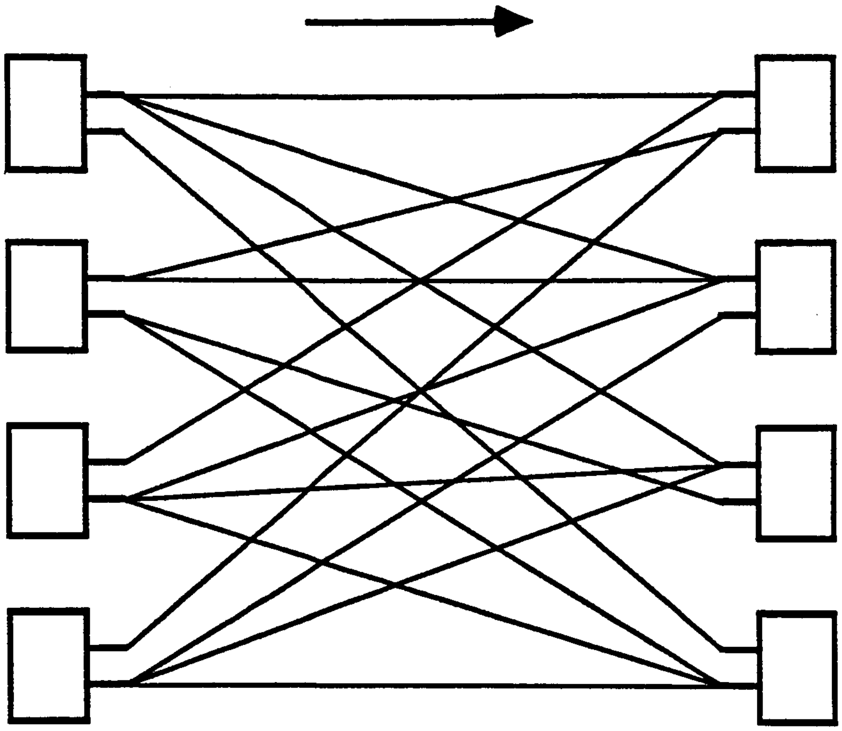


Fig. 2.9 Unidirectional, single-path $\mathcal{SBI}$ with standard stations but unequal degree, no grouping and overlapping subnetworks. $N = 4$; $C = 2$.

rates.

- For any problem that is formulated, the solution may be constrained to have some of the aforementioned attributes. For example, any $\mathcal{SBI}$ that employs frequency division as the sole means of permitting concurrency must have disjoint subnetworks.

2.2.4 Research Focus

From the sample problems, it is clear that a vast number of problems can be formulated. Rather than attempt to solve a multitude of specific problems, the primary thrust of this research has been to find a useful classification of $\mathcal{SBI}$'s and to understand the properties of several classes of regular $\mathcal{SBI}$'s. The computational complexity of solving certain design problems under the constraint that an $\mathcal{SBI}$ belong to a certain class has also been studied. Finally, Implementation issues pertaining to $\mathcal{SBI}$'s that belong to certain classes have been addressed, and are discussed in chapter 3.

2.2.5 Performance Measures

The primary performance measure used in this research is *throughput.* The throughput of an interconnection (for a given traffic pattern) is usually defined to be the rate of successful receptions, summed up over all stations, when the traffic pattern is adhered to. This definition of throughput is appropriate for single-destination traffic and for interconnections in which sending the same message to several destinations is the same as sending them different messages. However, in interconnections that employ shared media, the reception of a message by several stations may be a free byproduct of its reception by the destination. As will be seen later, in an $\mathcal{SBI}$ with overlapping subnetworks, a message may even be received by a station other than its destination in spite of the fact that its destination cannot receive it. As a result, there are three possible criteria for determining whether a reception of a message constitutes throughput.

(i) Destination-specific throughput. It is assumed that a message has a single destination, and it is considered to constitute throughput if and only if received by this destination. This is the most common definition of throughput.

(ii) Destination-independent throughput. A message constitutes throughput if received by some station. This applies if all the RS's are identical servers of some sort, and it doesn't matter which server receives a request.

(iii) Reception rate. Each successful reception constitutes throughput; i.e., the same message may be counted several times. This measure is appropriate only when there is an advantage to having a message received by multiple stations. The case of multi-destination messages is a combination of (i) and (iii).

The emphasis here will be on the first type of throughput, to which we refer simply as "throughput". Nevertheless, some results will also be presented for the other types. Since the comparison among various interconnections is largely independent of the access scheme that is being used, throughput will be represented by *concurrency*, which is the ratio of throughput to transmission rate, assuming an ideal access scheme and an infinite supply of messages. It should nevertheless be emphasized that, like throughput, the concurrency of an interconnection *is* a function of the traffic pattern; i.e., of the relative traffic level that is to be carried between each pair of stations. As a reference for the performance of $\mathcal{SBI}$, we note that, regardless of traffic pattern, the concurrency of $\mathcal{PBI}$ with $C \geq 1$ is C for the first two types and $N \cdot C$ for the third. A comparison of delay performance of $\mathcal{SBI}$, $\mathcal{PBI}$ and $\mathcal{SBB}$ will also be presented. In chapter 3, more will be said about the throughput with nonideal access schemes.

Another aspect of an interconnection's performance, which is of particular importance to fiber optic implementations, is the *power budget*. In this chapter, only the number of receivers that must be reached by a transmitter will be addressed. In the next chapter, other issues pertaining to power budget will be discussed, and it will be treated in more detail.

2.3 Unidirectional, Equal-Degree $\mathcal{SBI}$'s with Grouping and Disjoint Subnetworks

2.3.1 Characteristics

These $\mathcal{SBI}$'s are characterized by the fact that they consist of a collection of disjoint subnetworks, each of which connects a subset of the TS's with a subset of the RS's. All the TS's of a given subnetwork can reach all of its RS's through that subnetwork. All transmitters on a given subnetwork are heard by the same receivers, and transmitters on different subnetworks are heard by disjoint sets of receivers. Similarly, two receivers on the same subnetwork hear the same transmitters, and receivers on different subnetworks hear disjoint sets of transmitters. As a result, such $\mathcal{SBI}$'s can be implemented using bidirectional as well as unidirectional media. Furthermore, the separation between subnetworks need not be spatial; it can be in the frequency domain, polarization, and even in the time domain.

The basic configuration in this class is the single-path $\mathcal{SBI}$. We will explain how it is constructed and discuss its performance; then, various modifications will be proposed and evaluated.

2.3.2 Construction of the Single-Path $\mathcal{SBI}$

In order to illustrate the versatility of this $\mathcal{SBI}$, let us consider the problem of connecting N_T transmitting stations, each with C_T transmitters, to N_R receiving stations, each with C_R receivers. While $N_T{\neq}N_R$ is also possible with $\mathcal{PBI}$, $C_T{\neq}C_R$ is not; therefore, whenever comparing the two, it will be assumed that $C_T{=}C_R{=}C$. For convenience , it will also be assumed that

$$N_T = P \cdot C_R \qquad \text{and} \qquad N_R = Q \cdot C_T \tag{2.1}$$

where P and Q are integers.

To construct this $\mathcal{SBI}$, arrange the transmitting stations in C_R disjoint groups, each with P stations; similarly, arrange the receiving stations in C_T disjoint groups, each with Q stations. Next, construct $C_T \cdot C_R$ subnetworks, each connecting a group of transmitting stations to a group of receiving stations. Viewed differently, each transmitting station uses its jth transmitter to send messages to the jth group of receiving stations; similarly, each receiving station uses its ith receiver to receive messages from the ith group of transmitting stations. Fig. 2.10, which shows the transmitting and receiving stations at opposite ends of the drawing, represents a logic diagram of the connections; observe that each transmitting station has only one subnetwork in common with any given receiving station. Since there are $C_T \cdot C_R$ disjoint subnetworks, the degree of concurrency in this arrangement can reach the value $C_T \cdot C_R$. Finally, we note that this $\mathcal{SBI}$ reduces to well-known configurations in the following limiting cases:

a)$N_T = N_R = N$; $C_T = C_R = N - 1$. This corresponds to a fully connected topology with a point-to-point link from each transmitting station to each receiving station.

b) $C_T = C_R = 1$. This is a single broadcast bus.

2.3.3 Performance of the Single-Path $\mathcal{SBI}$

Unlike the concurrency with $\mathcal{PBI}$, which is always C, the concurrency provided by this $\mathcal{SBI}$, which can be as high as $C_T \cdot C_R$, depends on the traffic pattern. Therefore, any comparison between the two must state the traffic pattern to which it applies.

Uniform traffic pattern and single-destination transmissions

In this case, the $C_T \cdot C_R$ subnetworks of $\mathcal{SBI}$ can be treated as independent, identical subnetworks, each connecting P TS's to Q RS's; the throughput of $\mathcal{SBI}$

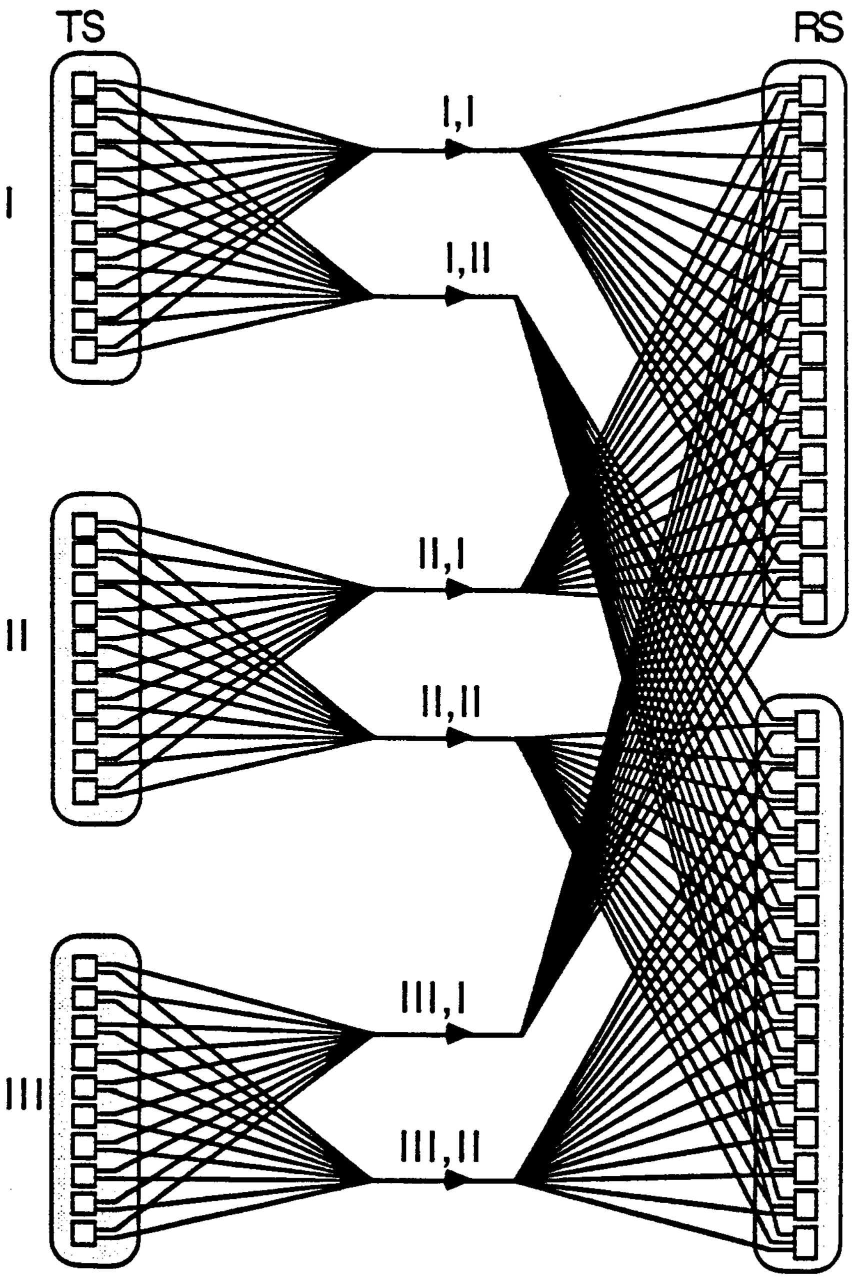


Fig. 2.10 Single-path, unidirectional, equal-degree $\mathcal{SBI}$ with disjoint subnetworks and grouping; $C_T \neq C_R$. ($N = 30$; $C_T = 2$; $C_R = 3$.)

can then be summarized by the expression

$$S^{\mathcal{SBI}} = K^{\mathcal{SBI}} \cdot C_T^{\mathcal{SBI}} \cdot C_R^{\mathcal{SBI}} \cdot B^{\mathcal{SBI}}. \tag{2.2}$$

The destination-independent throughput is the same as (2.2), and the maximum reception-rate is $N_R \cdot C_R$.

Since, for throughput purposes, $\mathcal{PBI}$ can always be treated as $C^{\mathcal{PBI}}$ independent, identical conventional broadcast networks, each connecting N_T transmitting stations to N_R receiving stations, its performance can be summarized by the expression

$$S^{\mathcal{PBI}} = K^{\mathcal{PBI}} \cdot C^{\mathcal{PBI}} \cdot B^{\mathcal{PBI}}. \tag{2.3}$$

K is a constant which depends on the channel access scheme ($0 < K \leq 1$). For the time being, ideal access schemes are assumed, so $K = 1$. To permit comparison, it is also assumed that $C_T{=}C_R$ and that N_T and N_R are the same in both systems. The above expressions can then be interpreted in several ways:

- With $C^{\mathcal{SBI}} = C^{\mathcal{PBI}} = C$ and $B^{\mathcal{SBI}} = B^{\mathcal{PBI}}$, the aggregate throughput of $\mathcal{SBI}$ is C times higher than of $\mathcal{PBI}$, since it increases quadratically rather than linearly with C.
- With $S^{\mathcal{SBI}} = S^{\mathcal{PBI}}$ and $C^{\mathcal{SBI}} = C^{\mathcal{PBI}} = C$, the transmission rate required with $\mathcal{SBI}$ is C times lower than that required with $\mathcal{PBI}$; i.e., slower (and cheaper) transmitters and receivers may be used for the same throughput.
- With $S^{\mathcal{SBI}} = S^{\mathcal{PBI}}$ and $B^{\mathcal{SBI}} = B^{\mathcal{PBI}}$, $C^{\mathcal{SBI}} = \sqrt{C^{\mathcal{PBI}}}$; i.e., $\mathcal{SBI}$ requires fewer transmitters and receivers.

Since each subnetwork of $\mathcal{SBI}$ serves only N/C transmitting stations and N/C receiving stations, as compared with N in $\mathcal{PBI}$, the average fraction of time that a subnetwork of $\mathcal{SBI}$ serves each of its member stations is higher by a factor of C than that fraction with $\mathcal{PBI}$. It follows that the average utilization of transmitters, of receivers and, in the case of fiber-optic implementations with a central wiring closet, of the fibers connecting stations with the wiring closet, is also higher by the same factor.

Multi-destination packets

Multicast to any subset of receiving stations that are connected to a given subnetwork is a byproduct of any successful transmission over that subnetwork. However, when several of a node's transmitters must transmit in order to reach the entire set of intended recipients, the performance of $\mathcal{SBI}$ degrades, and if transmissions by all C_T transmitters are required, as is the case for full broadcast, $\mathcal{SBI}$ loses its throughput advantage over $\mathcal{PBI}$. (With $\mathcal{PBI}$, multicast is always a free byproduct.)

Nonuniform traffic patterns

For nonuniform single-destination traffic patterns, the throughput with $\mathcal{SBI}$ may become as low as that with a single bus. This happens, for example, if all the traffic is from a single group of transmitting stations to a single group of receiving stations, in which case only one subnetwork can be used. We also note that, for any given source-destination pair, the maximum instantaneous data rate with $\mathcal{SBI}$ is B, as compared with $C \cdot B$ with $\mathcal{PBI}$.

2.3.4 Power Budget

An important aspect in which the single-path $\mathcal{SBI}$ outperforms $\mathcal{PBI}$ for any traffic pattern is the power splitting. While the use of $\mathcal{PBI}$ requires splitting the power of each transmitter N_R ways, it suffices to split it $N_R/C_T{=}Q$ ways for the single-path $\mathcal{SBI}$. Observe that if a station has C_T transmitters and is to be connected to N_R different receiving stations, the power of each transmitter must be split at least N_R/C_T ways. Consequently, $\mathcal{SBI}$ is optimal in this sense, and no other single-hop interconnection can do better. In the next chapter, power budget will be discussed in more detail.

2.3.5 Performance Tradeoff

The fact that the concurrency with the single-hop $\mathcal{SBI}$ can be as low as 1 and as high as C^2, depending on the traffic pattern, whereas that of $\mathcal{PBI}$ is always C, raises the question of whether one can do better than $\mathcal{PBI}$ without sacrificing flexibility. Lang, Valero and Fiol [28] addressed this question, with the assumption that a single station never does more than one thing at a time. Given C buses, they therefore consider performance not to be degraded as long as any C source-destination pairs, such that no source or destination appears more than once, can be accommodated concurrently. They have shown that

$$T_i \geq N_T - C + 1 \tag{2.4}$$

and

$$R_i \geq (N_T + N_R + 1) - (C + T_i), \tag{2.5}$$

where T_i and R_i are the number of transmitters and receivers on the ith bus, respectively. They have also shown that "minimal" configurations, i.e. those that achieve equality in (2.5), can be obtained with $T_i = N_T$. In this case, it is easy to see that the total number of receivers can be reduced by at most $C(C+1)$. Since the total number of receivers with $\mathcal{PBI}$ is $N_R \cdot C$, the fraction of receivers that can be saved, $(C+1)/N_R$, becomes negligible as the number of stations increases.

From the above results, it follows that there is a tradeoff between the maximum concurrency for a uniform traffic pattern, $\mathcal{C}_{\max}$, and the guaranteed concurrency, $\mathcal{C}_{\min}$; (for any pattern;) $\mathcal{PBI}$ and the single-path $\mathcal{SBI}$ are two extremes. We next present two parameterized compromises, both of which are equal-degree $\mathcal{SBI}$'s with grouping and disjoint subnetworks. In both cases, the guaranteed (minimum) concurrency is equal to the number of alternate paths between any two stations, and the maximum concurrency is equal to the number of disjoint subnetworks.

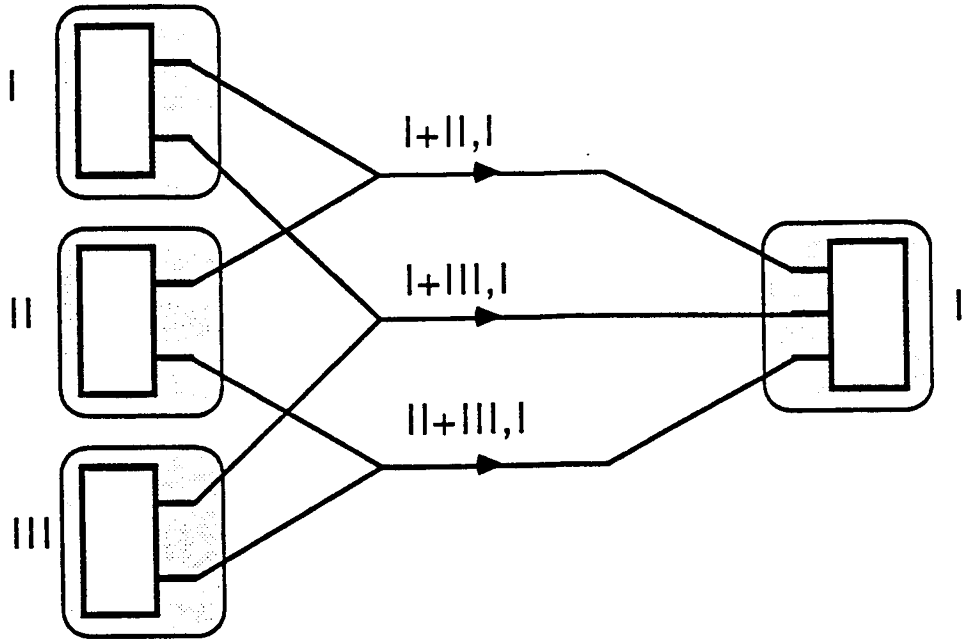


Fig. 2.11 Unidirectional, multiple-path, (two paths,) equal-degree $\mathcal{SBI}$ with disjoint subnetworks and grouping. (*SMP*.) A representative station is shown for each group. $G_T = 3$, $k_T = 2$, $G_R = k_R = 1$; $C_T = 2$, $C_R = 3$.

A single multiple-path $\mathcal{SBI}$ (*SMP*)

The sets of TS's and RS's are partitioned into G_T and G_R groups, respectively. A subnetwork is constructed to connect each possible combination of k_T groups of TS with each possible combination of k_R groups of RS. Fig. 2.11 depicts an *SMP*. The concurrency provided by an *SMP* is as follows:

$$\mathcal{C}_{\min} = \binom{G_T - 1}{k_T - 1} \cdot \binom{G_R - 1}{k_R - 1}; \qquad \mathcal{C}_{\max} = \binom{G_T}{k_T} \cdot \binom{G_R}{k_R};$$

$$C_T = \frac{k_T}{G_T} \cdot \binom{G_T}{k_T} \cdot \binom{G_R}{k_R}; \qquad C_R = \frac{k_R}{G_R} \cdot \binom{G_T}{k_T} \cdot \binom{G_R}{k_R};$$

$$\text{Power split} = k_R \cdot \frac{N_R}{G_R}. \tag{2.6}$$

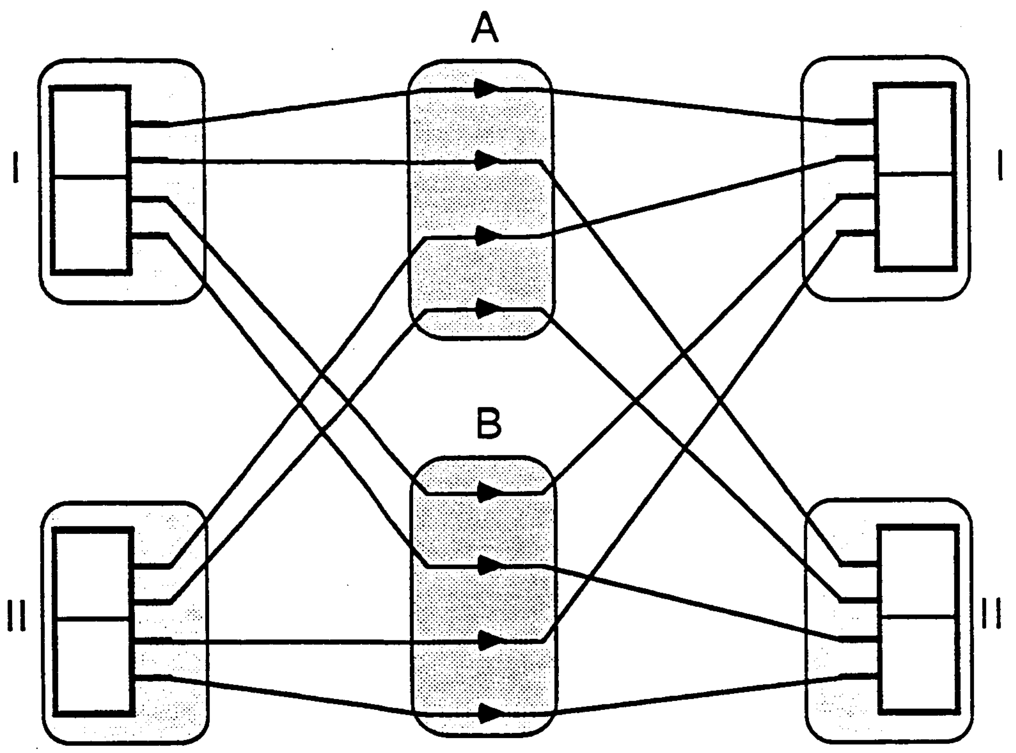


Fig. 2.12 Multiple single-path $\mathcal{SBI}$'s, each interconnecting all stations. (*MSP.*) A representative station is shown for each groups. $m = 2$; $C = 4$.

Multiple single-path $\mathcal{SBI}$'s (*MSP*)

m single-path $\mathcal{SBI}$'s are constructed, each of which utilizes 1/m of the transmitters and receivers. Fig. 2.12 depicts an *MSP*. The concurrency provided by an *SMP* is as follows:

$$\mathcal{C}_{\text{min}} = m; \qquad \mathcal{C}_{\text{max}} = \frac{C_T \cdot C_R}{m};$$

$$\text{Power split} = m \cdot \frac{N_R}{C_T}. \tag{2.7}$$

Comparison

To simplify the comparison, let $C_T = C_R = C$, $G_T = G_R = G$, $k_T = k_R = k$, and $N_T = N_R = N$. A comparison, conducted by equating C and $\mathcal{C}_{\text{min}}$ for the

C	$\mathcal{C}_{\min}$	$\mathcal{C}_{\max}$			Power Split		
		SMP	*MSP*	Hyb	*SMP*	*MSP*	Hyb
18	9	36	36	108	$\frac{N}{2}$	$\frac{N}{2}$	N

Table 2.1. Common feasible values of (C, $\mathcal{C}_{\min}$) for *MSP, SMP* and the Hybrid, and the resulting $\mathcal{C}_{\max}$ and power split. (Unidirectional).

two approaches and then comparing $\mathcal{C}_{\max}$ and the power split, shows that the performance is identical.

Although the identical performance suggests that the two approaches are perhaps different ways of describing the same interconnections, this is not the case. In fact, it can be shown that there are combinations of C and $\mathcal{C}_{\min}$ that are only feasible with one of the approaches. As an example, consider the case of $\mathcal{C}_{\min}$=36 and $C = 60$. With *SMP*, this can be achieved by letting $G = 5$ and $k = 3$. However, it is not feasible with *MSP*, since each subnetwork would utilize $\frac{60}{36}$ transmitters and receivers of each station... Table 2.1 presents C, $\mathcal{C}_{\min}$,$\mathcal{C}_{\max}$ and the power split with the different configurations for the only combination of $\mathcal{C}_{\min}$ and C, such that $1 < \mathcal{C}_{\min} < \mathcal{C}_{\max}$ and $C \leq 20$, which is feasible with all three configurations. Yet another approach involves the utilization of a fraction of the transmitters and receivers for the construction of a single-path $\mathcal{SBI}$; the remaining are used for the construction of a $\mathcal{PBI}$. It will be shown to outperform the two previous approaches, but it should be remembered that this is not an equal-degree $\mathcal{SBI}$. (A transmitter that is used in the $\mathcal{PBI}$ portion must reach N_R receivers, whereas one that is in the $\mathcal{SBI}$ portion reaches fewer receivers.)

A hybrid $\mathcal{SBI}$ -$\mathcal{PBI}$ interconnection

C' transmitters and C' receivers of each station are are used for a $\mathcal{PBI}$, and the

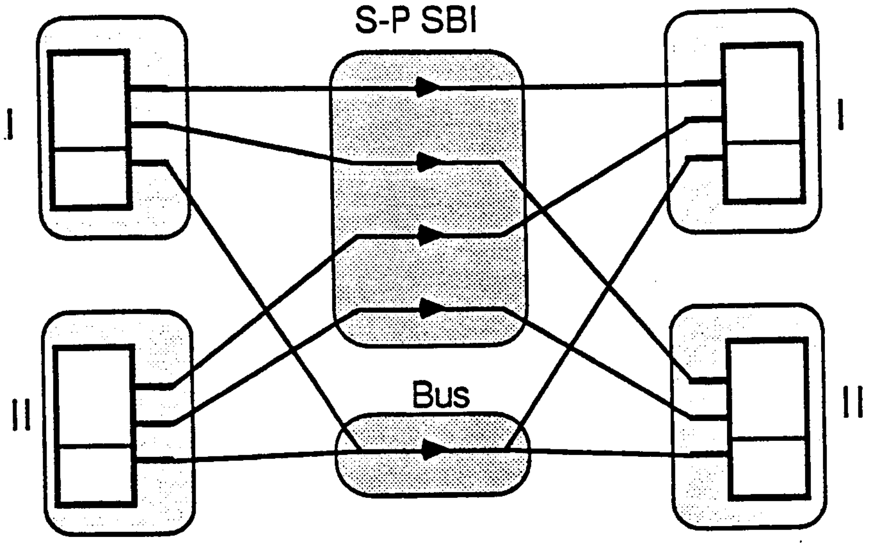


Fig. 2.13 Unidirectional hybrid $\mathcal{SBI}$-$\mathcal{PBI}$. $C = 3,\ C' = 1$.

remaining ones are used for a single-path $\mathcal{SBI}$. An example of such an $\mathcal{SBI}$ is depicted in Fig. 2.13. The performance is as follows:

$$\mathcal{C}_{\text{min}} = C' + 1; \qquad \mathcal{C}_{\text{max}} = C' + (C_T - C') \cdot (C_R - C');$$

$$\text{Power split (worst case) } = N_R. \tag{2.8}$$

To compare this with the two previous approaches, let us again assume equal $\mathcal{C}_{\text{min}}$ and C and compute $\mathcal{C}_{\text{max}}$. Using *MSP* terminology, the hybrid configuration has

$$\mathcal{C}_{\text{max}} = (C - m + 1)^2 + m - 1, \tag{2.9}$$

as compared with C^2/m for *SMP* and *MSP*. It can be shown that the performance is equal if $m = C$ or $m = 1$, and the hybrid performs better in all other cases. (This is proved by showing that, for any given C, the difference is zero only at two values of m, namely 1 and C, and that for $m = C/2$ the hybrid is always superior.)

Furthermore, the performance advantage of the hybrid increases as C increases, for any fixed value of m.

For the case wherein $C_T = C_R = C$, another way of describing the allocation of transmitters and receivers to the two components of the hybrid $\mathcal{SBI}$ is as a fraction of C: $\alpha \cdot C$ transmitters and receivers of each station are used for a single-path $\mathcal{SBI}$, and the remaining ones are used to construct a $\mathcal{PBI}$. The performance is then given by

$$\mathcal{C}_{\text{min}} = (1-\alpha) \cdot C + 1 = \Omega(C); \qquad \mathcal{C}_{\text{max}} = (1-\alpha) \cdot C + \alpha^2 \cdot C^2 = \Omega(C^2). \quad (2.10)$$

The performance of the hybrid thus incorporates the advantages of the two constituent configurations, up to a constant factor.

For the sake of completeness, it should be noted that the guaranteed concurrency of *MSP* can sometimes be improved by permuting the station numbers in the different constituent single-hop $\mathcal{SBI}$'s. However, this violates the grouping constraint; also, C must be sufficiently large, so that no two stations are in the same group in all constituent single-path $\mathcal{SBI}$'s.

The hybrid has another advantage, which is the flexibility in the allocation of hardware to the two components. This is illustrated in Fig. 2.14, which shows, for each of the three configurations, all feasible combinations of $\mathcal{C}_{\text{min}}$ and $\mathcal{C}_{\text{max}}$ subject to the constraint that $1 < \mathcal{C}_{\text{min}} < \mathcal{C}_{\text{max}}$ and $C \leq 20$. One can also see a significant advantage of *MSP* over *SMP* in this respect. Triangles, boxes and plus signs correspond to *SMP, MSP* and the hybrid, respectively.

2.4 Bidirectional Equal-Degree $\mathcal{SBI}$'s With Grouping

A *bidirectional* $\mathcal{SBI}$ is an $\mathcal{SBI}$ which consists of disjoint subnetworks, such that station i can reach station j over a given subnetwork if and only if station j can

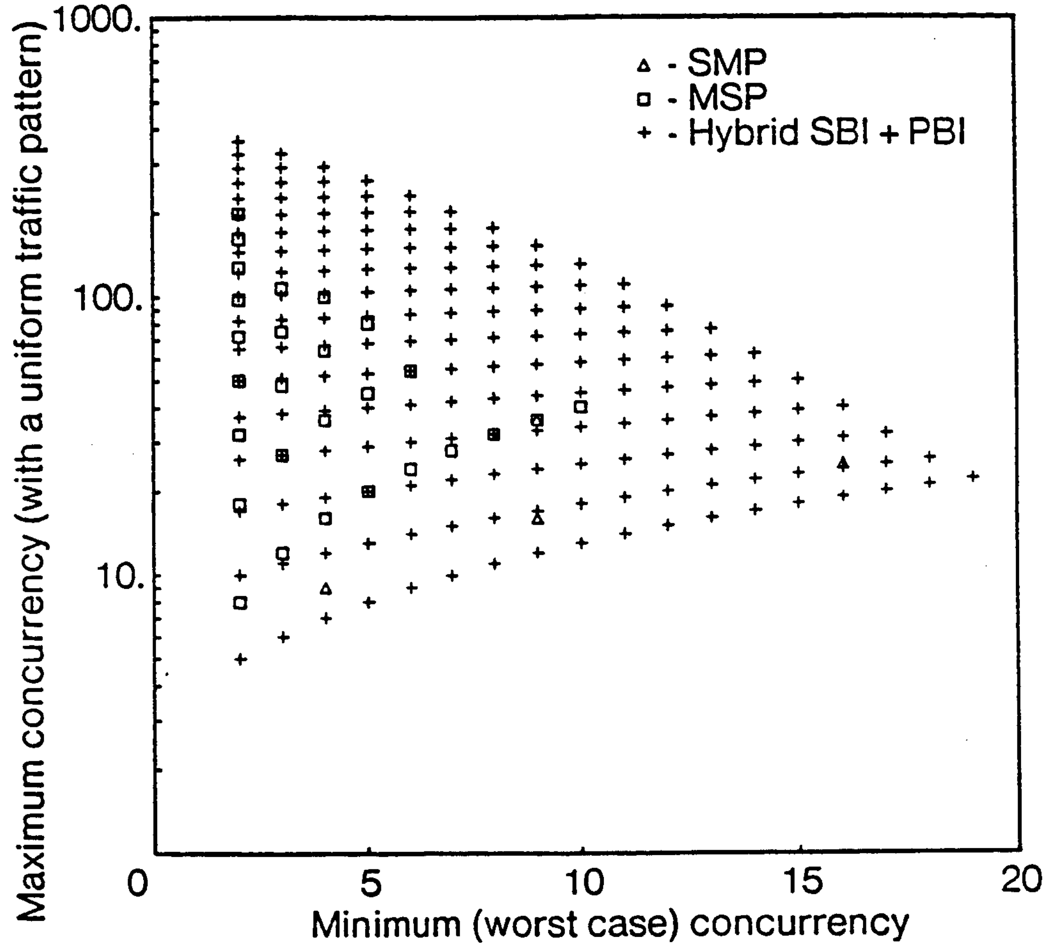


Fig. 2.14 Feasible ($C_{\min}$,$C_{\max}$) combinations with unidirectional *SMP*. *MSP*. and a hybrid.

reach station i over the *same* subnetwork. In other words, a bidirectional $\mathcal{SBI}$ is a collection of disjoint subnetworks, each of which provides bidirectional communication among a subset of the stations using one of their transmitters and one of their receivers. It should be noted that a bidirectional $\mathcal{SBI}$ can be implemented using bidirectional as well as unidirectional media. However, if transceivers are to be used, the media must be bidirectional. Since each subnetwork is identical to a broadcast bus, (interconnecting only a subset of the stations,) the bidirectional $\mathcal{SBI}$ can be operated in conjunction with any access scheme. This will be elaborated upon in chapter 3.

2.4.1 Design Space and Graph Representation

A station of a bidirectional $\mathcal{SBI}$ must clearly have an equal number of transmitters and receivers; for convenience, we think of them as transceivers. The interconnection designer must allocate transceivers to stations, and then assign transceivers to subnetworks so as to provide single-hop connectivity among the stations.

A bidirectional $\mathcal{SBI}$ can be modeled as an undirected graph, with C_i nodes representing the ith station. (One node per transceiver.) There is an edge between two nodes if and only if the corresponding transceivers can hear each other. Alternatively, it can be modeled as a directed graph.

2.4.2 Bidirectional $\mathcal{SBI}$ for Maximum Throughput with a Uniform Traffic Pattern

From symmetry considerations, it is obvious that all stations should be equipped with an equal number of transceivers; we denote it by C. It is also clear that maximum throughput will be attained with a single-path $\mathcal{SBI}$. Finally, we note that all bidirectional $\mathcal{SBI}$'s have disjoint subnetworks, and arrive at the following construction rule: divide the stations into $(C + 1)$ groups of equal size. Next, construct a subnetwork for each pair of groups; each such subnetwork interconnects all stations in both groups. The number of subnetworks is

$$\frac{C \cdot (C+1)}{2}, \qquad (2.11)$$

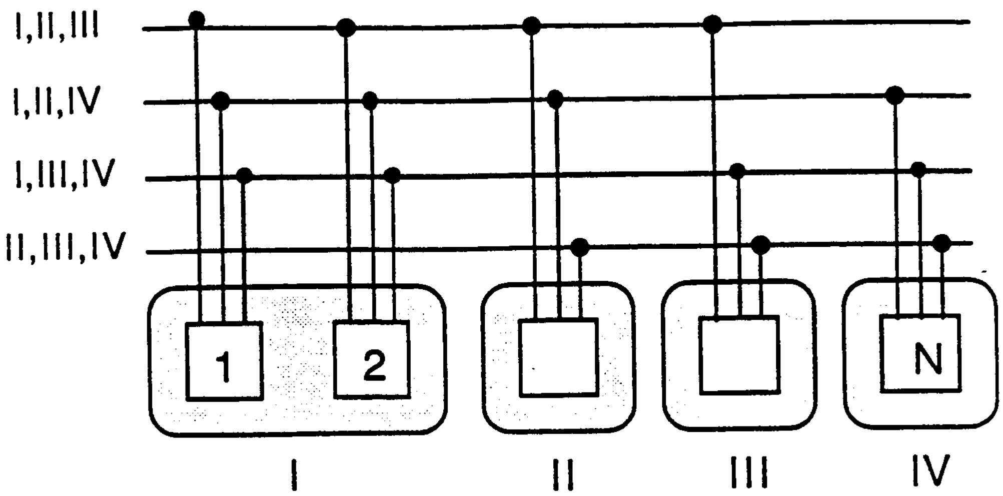


Fig. 2.15 Bidirectional *SMP*. $G = 4$, $k = 3$, $C = 3$.

and each station is a member of exactly C of them. We observe that the bidirectional, single-path $\mathcal{SBI}$ provides C paths between two stations that are members of the same group. In fact, a bidirectional $\mathcal{SBI}$ with grouping can always be described as a (k, C) $\mathcal{SBI}$, providing k paths between any two stations in different groups and C paths between stations in the same group. A $(1,3)$ $\mathcal{SBI}$ is depicted in Fig. 2.3.

2.4.3 Performance Tradeoff

For nonuniform traffic patterns, the performance tradeoff here is similar to the one encountered in the single-path unidirectional $\mathcal{SBI}$. As in the unidirectional case, three parameterized approaches are explored.

A Single Multiple-Path, Equal-Degree Bidirectional $\mathcal{SBI}$ (*SMP*)

This $\mathcal{SBI}$ is constructed in a similar manner to the single-path one, except that now each subnetwork interconnects stations of $k \geq 2$ groups. G is again used to denote the number of groups. Fig. 2.15 depicts such an $\mathcal{SBI}$. The maximum concurrency, $\mathcal{C}_{\max}$, which is achieved for a uniform traffic pattern, is the number of subnetworks that can be constructed; the minimum concurrency, $\mathcal{C}_{\min}$, is equal to the number of subnetworks that any two groups have in common. The following equations relate the various parameters:

$$\mathcal{C}_{\min} = \binom{G-2}{k-2}; \quad \mathcal{C}_{\max} = \binom{G}{k}; \quad C = \frac{k \cdot \binom{G}{k}}{G}.$$

$$\text{Power split } = k \cdot \frac{N}{G}. \tag{2.12}$$

Multiple $(1, C/m)$, Equal-Degree Bidirectional $\mathcal{SBI}$'s (*MSP*)

These are simply m identical $(1, C/m)$ bidirectional $\mathcal{SBI}$'s, each employing $\frac{C}{m}$ transceivers per station. The performance is

$$\mathcal{C}_{\min} = m; \qquad \mathcal{C}_{\max} = \frac{C \cdot (C+m)}{2m};$$

$$\text{Power split } = \frac{2 \cdot m \cdot N}{C}. \tag{2.13}$$

This holds for values of m which divide C. Fig. 2.16 depicts such an $\mathcal{SBI}$.

Hybrid $\mathcal{SBI}$ -$\mathcal{PBI}$ interconnection

As was the case with the unidirectional $\mathcal{SBI}$, C' transmitters and receivers of each station are used to construct C' parallel buses, and the remaining ones are used to construct a $(1, C - C')$ bidirectional, equal-degree $\mathcal{SBI}$. Again, it is important to note that this hybrid $\mathcal{SBI}$ is not equal-degree. The performance with the hybrid configuration is

$$\mathcal{C}_{\min} = C' + 1; \qquad \mathcal{C}_{\max} = C' + (C - C') \cdot (C - C' + 1)/2;$$

$$\text{Power split (worst case) } = N. \tag{2.14}$$

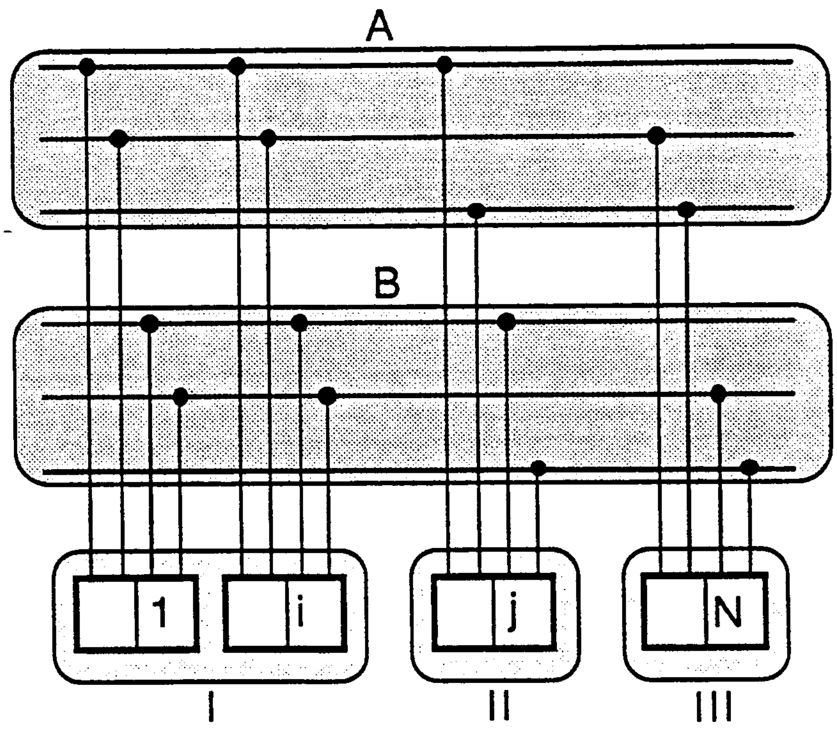


Fig. 2.16 Bidirectional *MSP*. $m = 2, C = 4$.

Comparison

Let us again equate C and the worst case concurrency ($\mathcal{C}_{\text{min}}$), and compare the resulting maximum concurrency ($\mathcal{C}_{\text{max}}$) as well as the power split. For convenience in analysis, k and G are used as the independent variables, and C, $\mathcal{C}_{\text{max}}$ and the power split are expressed in terms of those.

SMP

$$\mathcal{C}_{\text{max}} = \binom{G}{k}; \quad \text{Power split } = \frac{k}{G} \cdot N. \tag{2.15}$$

MSP

$$m = \mathcal{C}_{\min} = \binom{G-2}{k-2}; \qquad C = \frac{k}{G} \cdot \binom{G}{k};$$

$$\mathcal{C}_{\max} = \binom{G}{k} \cdot \frac{k}{2G} \cdot \left(\frac{G-1}{k-1} + 1\right). \tag{2.16}$$

Therefore,

$$\left(0.5 + \frac{k}{2G}\right) \cdot \mathcal{C}_{max}^{SMP} \leq \mathcal{C}_{max}^{MSP} \leq \mathcal{C}_{max}^{SMP}, \tag{2.17}$$

with equality only in the case that both reduce to a single bus ($G = k$) or to a single single-path $\mathcal{SBI}$ ($k = 2$). As for power split:

$$\text{Power split:} \quad \frac{2(k-1)}{G-1} \cdot N. \quad (\geq \frac{k}{G} \cdot N.) \tag{2.18}$$

The bidirectional *SMP* thus also provides a better power split than the bidirectional *MSP*.

Hybrid

$$C = \frac{k}{G} \cdot \binom{G}{k}; \qquad C' = \mathcal{C}_{\min} - 1 = \binom{G-2}{k-2} - 1;$$

$$\mathcal{C}_{\max} = \binom{G}{k} \cdot \frac{k}{2 \cdot G \cdot (G-1)} \left[\frac{(G-k)^2}{k-1} \cdot \binom{G-2}{k-2} + 3 \cdot G - k - 2\right]. \tag{2.19}$$

It can be shown that the hybrid outperforms the *SMP*, with equality only when $k = 2$, $G-1$, or G. The equality can be shown by direct substitution. The inequality in all other cases is shown as follows: by simple manipulations and factorization of terms, the condition for inequality reduces to

$$\binom{G-2}{k-2} \geq \frac{(k-1)(2G-k-2)}{k \cdot (G-k)}.$$

The right hand side is monotonically increasing with k, and assumes its maximum finite value at $k = G-1$, which yields $G-2$. The left hand side assumes values that are smaller than or equal to $G-2$ only when $k = 1, 2, 3, G-1$ or G. Substituting these values, we see that $k = 1$ is not feasible, $k = 2$ and $k = G-1$ yield equality, and $k = 3$ yields an inequality. $k = G$ was shown to yield equality in (2.19). This completes the proof.

The hybrid thus outperforms both *MSP* and *SMP*, with equality only in extreme cases. As in the unidirectional case, the hybrid approach provides guaranteed concurrency which is linearly proportional to C as well as a maximum concurrency which grows quadratically with C. However, in both cases there is a tradeoff between performance and power budget. The two equal-degree approaches are quite similar in both respects; the $\mathcal{SBI}$ $+\mathcal{PBI}$ can be significantly better in performance, but is significantly worse in terms of power budget. The three approaches also differ in the degree of flexibility that is provided to the designer. In the *SMP*, G and k must satisfy $\frac{k}{G} \cdot \binom{G}{k} = C$; in *MSP*, m must divide C; the least restrictive is $\mathcal{SBI}$ $+\mathcal{PBI}$, in which any number of transceivers (0...C) may be used for the $\mathcal{PBI}$. This is illustrated in Fig. 2.17 which shows, for each of the three configurations, all feasible combinations of $\mathcal{C}_{\min}$ and $\mathcal{C}_{\max}$ subject to the constraint that $1 < \mathcal{C}_{\min} < \mathcal{C}_{\max}$ and $C \leq 20$. Triangles, boxes and plus signs correspond to *SMP*, *MSP* and the hybrid, respectively.

Table 2.2 presents C, $\mathcal{C}_{\min}$, $\mathcal{C}_{\max}$ and the power split with the different configurations for all combinations of $\mathcal{C}_{\min}$ and C, such that $1<\mathcal{C}_{\min}<\mathcal{C}_{\max}$ and $C \leq 20$, which are feasible with all three configurations. Fig. 2.17 and table 2.2 both suggest that in practice, when it is desired to achieve a certain combination of $\mathcal{C}_{\min}$ and $\mathcal{C}_{\max}$, the choice between *SMP* and *MSP* may depend primarily on feasibility.

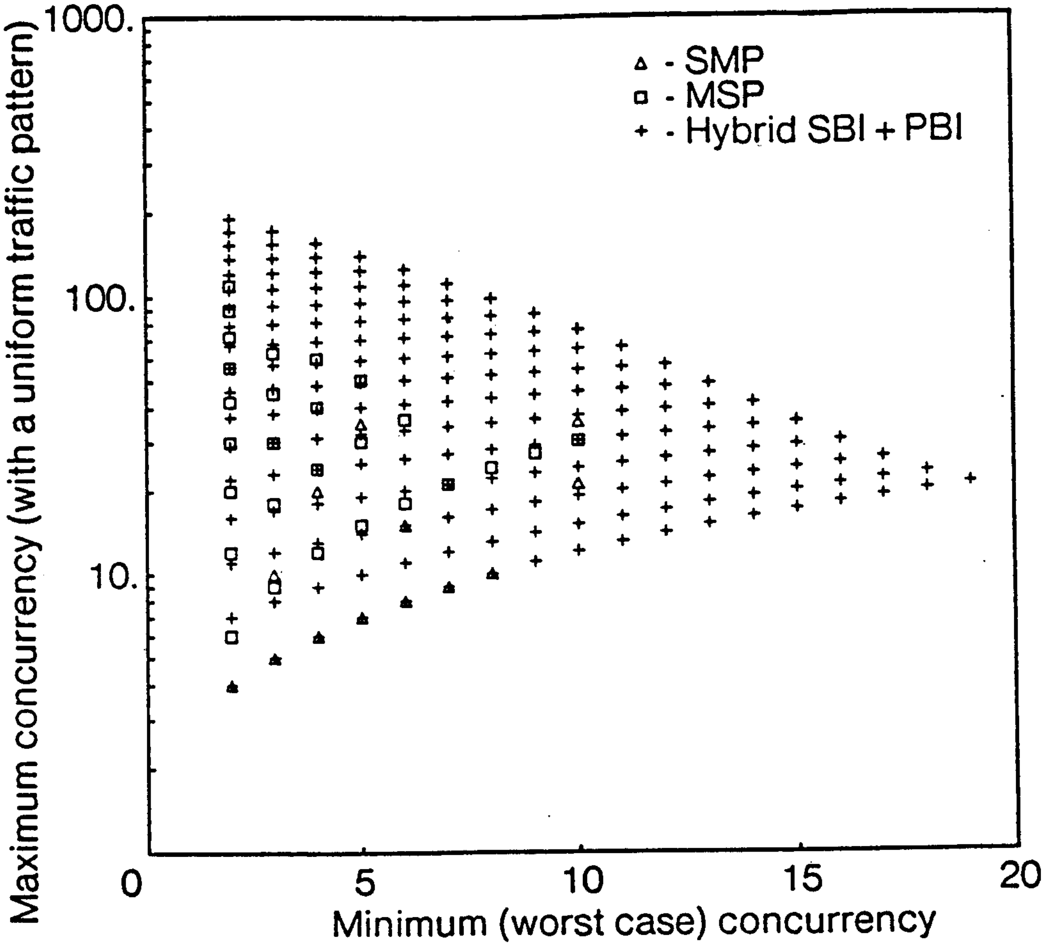


Fig. 2.17 Feasible ($\mathcal{C}_{\min}$,$\mathcal{C}_{\max}$) combinations with bidirectional *SMP*, *MSP*, and a hybrid.

2.4.4 Relationship between the Unidirectional and Bidirectional $\mathcal{SBI}$'s

In order to convince the reader that the unidirectional and bidirectional $\mathcal{SBI}$'s are not unrelated, we now explain how the $(1, C)$ bidirectional $\mathcal{SBI}$ can be obtained from a single-path unidirectional $\mathcal{SBI}$ with $C + 1$ transmitters and receivers per station. In the unidirectional $\mathcal{SBI}$, subnetwork (i, j), $1 \leq i, j \leq C + 1$, connects

C	$\mathcal{C}_{\min}$	$\mathcal{C}_{\max}$			Power Split		
		SMP	*MSP*	Hyb	*SMP*	*MSP*	Hyb
6	3	10	9	12	$\frac{3}{5}N$	N	N
15	5	35	30	70	$\frac{3}{7}N$	$\frac{2N}{3}$	N
20	10	35	30	75	$\frac{4}{7}N$	N	N

Table 2.2. Common feasible values of (C, $\mathcal{C}_{\min}$) for *MSP, SMP* and the Hybrid, as well as the resulting $\mathcal{C}_{\max}$ and power split. (Bidirectional.)

the ith group of TS's to the jth group of RS's. To obtain the bidirectional $\mathcal{SBI}$, combine subnetwork (i, j) with (j, i) whenever $j \neq i$. Since each subnetwork now interconnects all member stations in both directions, remove subnetworks (i, i) and do away with one transmitter and one receiver per station. The result is a $(1, C)$ bidirectional $\mathcal{SBI}$.

2.5 Concurrency with Unidirectional Broadcast Media

2.5.1 A Graph Model for Unidirectional Media

Any interconnection that uses unidirectional media can be described as a directed acyclic graph, (DAG,) in which source nodes (indegree zero) represent transmitters, destination nodes (outdegree zero) represent receivers, and there is a path from node i to node j if and only if receiver j can hear transmissions of transmitter i. Contrary to their role as switches in networks employing *point-to-point* links, intermediate nodes in unidirectional *broadcast* media act as *directional couplers*: a signal which appears on one of the inbound edges exits over all outbound edges. Each transmitter thus spans a tree in the graph, (of which it is the root), and a transmission is heard by all the leaves of the transmitter's tree. It should be noted that trees rooted at different transmitters may nevertheless have some common nodes and edges.

2.5.2 Interference – A Criterion for Concurrency

Assuming that a receiver can receive a message if and only if that is the only message that it hears, two paths in an interconnection employing *point-to-point* links must be edge-disjoint in order for successful transmissions to take place over them concurrently. (Node-disjointness may or may not be required, depending on the architecture of the intermediate nodes.) However, in an interconnection employing *broadcast* media, a stronger condition must be met: for two successful transmissions to take place concurrently, the recipient of one transmission must not be in the tree spanned by the transmitter of the other transmission.

To facilitate the determination of the concurrency provided by a graph, we next present a relationship between two paths in a graph, which can serve as criterion for determining whether or not concurrent successful transmissions can take place over them.

Definition. path (A, B) in a directed graph *interferes* with path (C, D) if and only if there is a path (A, D). The interference of path a with path b is denoted by $I < a, b >$. Two paths can carry concurrent successful transmissions if and only if they are mutually noninterfering.

If one is interested only in paths between source nodes and destination nodes, as we are, the general directed graph can be replaced with an equivalent directed

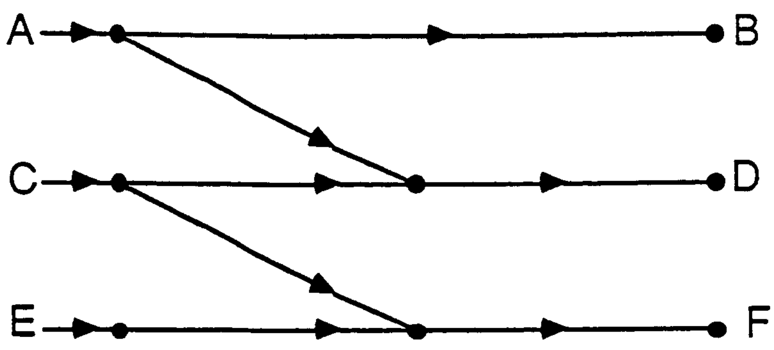


Fig. 2.18 Interference in a directed graph. (A, B) interferes with (C, D), and (C, D) interferes with (E, F). However, (C, D) does not interfere with (A, B), (not commutative,) and (A, B) does not interfere with (E, F). (Not transitive.)

bipartite graph; each source-destination path in the original graph is represented by a source-destination edge in the bipartite graph, and parallel edges are then consolidated.

Theorem 2.1. Interference has the following properties:

1) In general, it is not commutative and not transitive.

2) $I < (A,B),(C,D) >$ in a DAG G if and only if $I < (D,C),(B,A) >$ in the DAG G' which is obtained from G by reversing all of its edges.

3) In a graph with commutative interference, interference is also transitive.

4) If, in a given bipartite graph G, interference is transitive, and all source nodes have equal degrees, then interference is also commutative in that graph.

Proof

1) In Fig. 2.18, for example, $I < (A,B),(C,D) >$ and $I < (C,D),(E,F) >$; however, (C, D) does not interfere with (A, B) (not commutative) and (A, B) does not interfere with (E, F) (not transitive).

2) $I < (A,B),(C,D) > \Rightarrow \exists(A,D)$ in G. Therefore, $\exists(D,A)$ in G', and hence

$I < (D, C), (B, A) >$. Similarly, $I < (D, C), (B, A) > \Rightarrow I < (A, B), (C, D) >$.

Corollary. paths (A, B) and (C, D) in G are mutually noninterfering if and only if, in G', (B, A) and (D, C) are mutually noninterfering. Graphs G and G' thus have the same concurrency properties. This will be elaborated upon in chapter 3.

3) Let $\Gamma(A)$ be the set of vertices to which there is a path from A.

   (a) Commutative interference $\Rightarrow (I < (A, B), (C, D) > \Leftrightarrow \Gamma(A) = \Gamma(C))$.

   Proof. $I < (A, B), (C, D) > \Rightarrow$ for all $i \in \Gamma(A)$, $I < (A, i), (C, D) >$. Due to commutativity, this implies that $I < (C, D), (A, i) >$, and thus that $\Gamma(A) \subseteq \Gamma(C)$. Similarly, since $I < (A, B), (C, D) > \Rightarrow I < (C, D), (A, B) >$, it follows that $\Gamma(C) \subseteq \Gamma(A)$. Consequently, $\Gamma(A) = \Gamma(C)$.

   The reverse direction is trivial.

   (b) From (a) it follows that the commutativity causes the transitivity of interference to be equivalent to that of equality, thus completing the proof.

4) Transitivity implies that $(I < (A, B), (C, D) >$ and $I < (C, D), (E, F) >) \Rightarrow I < (A, B), (E, F) >$. Therefore, $I < (A, B), (C, D) > \Rightarrow \Gamma(A) \supseteq \Gamma(C)$. Since $||\Gamma(A)|| = ||\Gamma(C)||$, (equal degree,) it immediately follows that $\Gamma(A) = \Gamma(C)$. This, in turn, implies that $I < (C, D), (A, B) >$. □

2.5.3 Determining the Maximum Concurrency of a Given Graph

An important attribute of a directed graph is the maximum number of concurrent successful transmissions that it can support; i.e., the maximal set of mutually noninterfering edges in the corresponding bipartite graph. (In "conventional" graph-theoretic terms, we wish to find the maximal set of vertices such that the subgraph induced by them is a perfect matching.)

Proposition 2.2. the problem of finding the maximal set of mutually noninterfering edges in a given bipartite graph, and the problem of finding the cardinality of that set, are both NP-complete in the number of vertices.

Proof

a) Reduction from maximal independent set (MIS)[29]

MIS. An independent set in a graph $G = (V, E)$ is a subset $V' \subseteq V$ such that, for all $u, v \in V'$, the edge (u, v) is not in E. The independent set problem asks, for a given graph $G = (V, E)$ and a positive integer $J \leq |V|$, whether G contains an independent set V' having $|V'| \geq J$. We will use an equivalent version, namely that of determining the largest value of J such that the answer to the original one is "yes".

Given an instance (V, E) of MIS with $||V|| = N$, construct a bipartite graph as follows: on the left side, place $3N$ vertices, designated $A_1..A_N$, $B_1..B_N$, $C_1..C_N$. On the right side, place $5N$ vertices, designated $A'_1..A'_N$, $B'_1, B''_1..B'_N, B''_N$, $C'_1, C''_1..C'_N$, C''_N. For all $1 \leq i \leq N$, connect A_i with A'_i, with B''_i and with C''_i; next, connect B_i with B'_i and with B''_i; finally, connect C_i with C'_i and with C''_i. This completes the input-independent part of the construction. Now, for all (i, j), $i \neq j$, connect A_i with B'_j and with C'_j if and only if $(V_i, V_j) \in E$. Fig. 2.19 depicts a sample instance of MIS along with the corresponding instance of MNIE. The thick edges correspond to the input-dependent portion of the construction.

Claim. the maximum number of mutually noninterfering edges, $||MNIE||$, is $2N + ||MIS||$. Also, $\{V_i : A_i$ is the source of an edge in the maximal set of mutually noninterfering edges$\} =$ MIS.

Proof.

(1) $||\text{MNIE}|| \geq 2N + ||\text{MIS}||$. Given the instance of MIS along with the solution, we construct a set of mutually-noninterfering edges in the corresponding instance

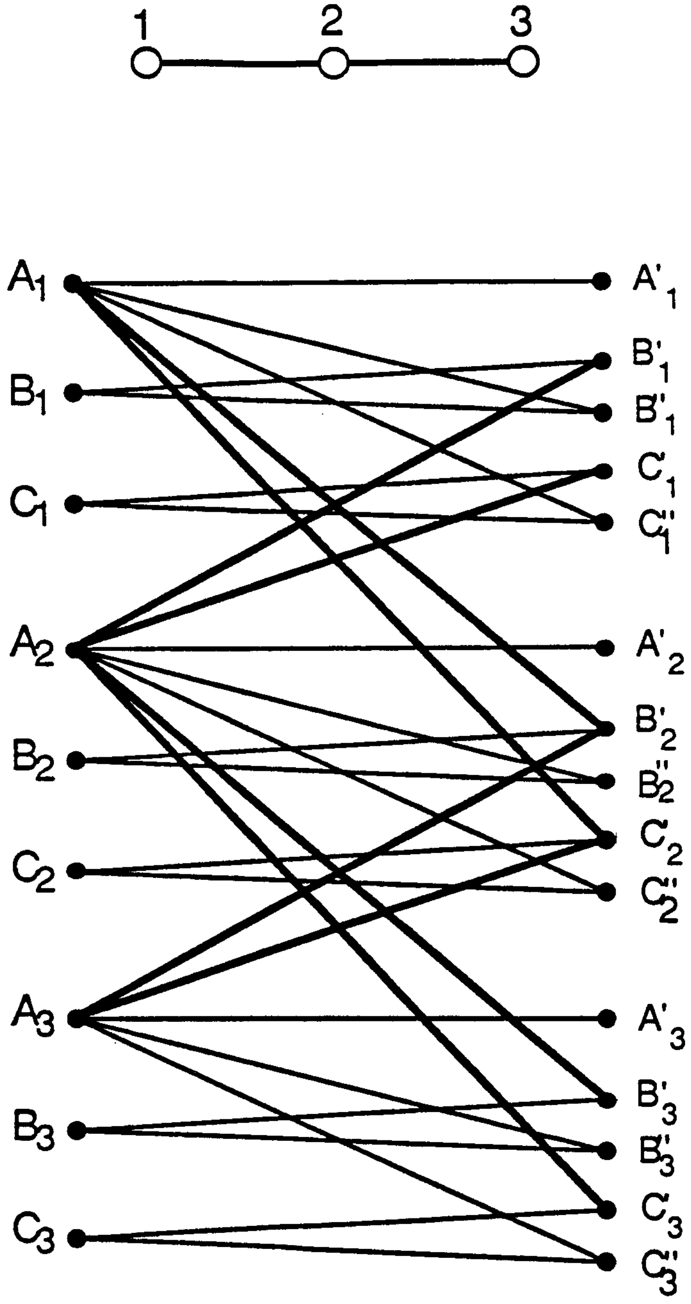


Fig. 2.19 Reduction from MIS (top) to MNIE (bottom). MIS=$\{1,3\}$. MNIE= $\{(A_1,A_1'),\ (B_1,B_1'),\ (C_1,C_1'),\ (A_3,A_3'),\ (B_3,B_3'),\ (C_3,C_3'),\ (B_2,B_2''),\ (C_2,C_2'')\}$.

of MNIE as follows. For $\{i : V_i \in \text{MIS}\}$, select (A_i, A_i'), (B_i, B_i') and (C_i, C_i').

For $\{i : V_i \notin \text{MIS}\}$, select (B_i, B_i'') and (C_i, C_i''). To see that these edges are mutually noninterfering, we consider each type separately. Edges of type (B_i, B_i'') and (C_i, C_i'') can only be interfered with by an edge whose source is A_i. Since no edges involving A_i were chosen for i such that $V_i \notin$ MIS, the edges of those two types are not interfered with. Since these edges can only interfere with (B_i, B_i') and (C_i, C_i'), respectively, and those were not chosen for vertices not in MIS, there is no problem. As for vertices in MIS, all the edges corresponding to them begin and terminate at the same i, or at single-primed nodes representing vertices not in MIN. However, no edges terminating at such nodes were selected, so there is no interference.

(2) Any solution that involves A_i and A_j, such that $(V_i, V_j) \in \{E\}$, can be improved upon by not using A_i. This is so because if both A_i and A_j are used, it follows that B_i and C_i cannot be used. If A_i is not used, (B_i, B_i'') and (C_i, C_i'') can be used; this results in an increase of one in the number of selected edges.

We conclude that any locally optimal selection will appear to have been constructed as explained in (1), with MIS replaced by some independent set. It follows immediately that the global optimum is indeed (1). It is also obvious from the construction that MIS= $\{i :\ A_i$ is the source of an edge in the maximal set of mutually noninterfering edges$\}$.

b) Reduction to Maximum Clique. For each edge in a given instance of MNIE, construct a vertex in the corresponding instance of maximum clique. Next, connect V_i and V_j if and only if the corresponding edges in MNIE are mutually noninterfering. The proof is trivial.

2.5.4 Throughput of an Equal-Degree Bipartite Graph for Randomized Transmissions with Random Destinations

Consider a bipartite graph whose vertices are T transmitters with outdegree Q and R receivers with indegree P. A slotted time system is assumed. In each time slot, each transmitter transmits with probability p. Whenever it transmits, the destination is chosen at random and with equal probabilities from among the Q candidates. The transmission process is independent from transmitter to transmitter and from slot to slot. We refer to this as a Bernoulli (p) process.

Proposition 2.3. The maximum type-1 throughput of any such graph (maximized over p) is at least $\frac{1}{e} \cdot \frac{R}{Q}$. (It increases as P decreases; for $P = 2$ it becomes $0.5 \cdot \frac{R}{Q}$.)

Proof

$$\Pr\{\text{a given receiver receives a transmission in a given time slot}\} = \tag{2.20}$$
$$= P \cdot \frac{p}{Q}(1-p)^{P-1}$$

The aggregate throughput is obtained by multiplying this by the total number of receivers, R. It is maximized by setting $p = \frac{1}{P}$, yielding

$$S_{max} = \frac{R}{Q} \cdot (1 - \frac{1}{P})^{P-1}. \tag{2.21}$$

Therefore,

$$\frac{1}{e} \cdot \frac{R}{Q} \leq S_{max} \leq 0.5 \cdot \frac{R}{Q}, \quad C \geq 2. \tag{2.22}$$

Corollary. With an unslotted system (pure ALOHA), $S_{max} \geq \frac{1}{2e} \cdot \frac{R}{Q}$.

2.5.5 Maximum Concurrency of an Equal-Degree Bipartite Graph

Consider bipartite digraphs with T source nodes (transmitters), each with outdegree Q, and R destination nodes (receivers), each with indegree P. Parallel edges are not allowed.

Proposition 2.4. The maximum number of concurrent successful transmissions, i.e., transmissions that are received concurrently by their destinations, (maximum over all such graphs and over all source-destination combinations,) is

$$\text{Concurrency} \leq \min\left\{ \left\lfloor \frac{\sqrt{T \cdot Q \cdot R \cdot P}}{P + Q - 1} \right\rfloor , \sqrt{T \cdot Q \cdot R \cdot P} - P \cdot Q + 1 \right\}. \tag{2.23}$$

Proof. Let us denote the transmitters by $\{T_i : i = 1, 2, ..., T\}$ and the receivers by $\{R_i : i = 1, 2, ..., R\}$. Let $\Gamma(T_i)$ denote the set of receivers that can hear T_i and let $\Gamma^{-1}(R_i)$ denote the set of transmitters that can be heard by R_i.

Suppose that there is a successful transmission from T_i to R_j; edge (i, j) in the bipartite graph is then said to be in state S (for success). It follows that all edges $\{(i, l) : l \in \Gamma(i), l \neq j\}$ are carrying redundant information; i.e., they are active but cannot constitute successes; their state is denoted by RED. Since those edges are active, all edges $\{(m, l) : m \in \Gamma^{-1}(l), m \neq i\}$ are indirectly redundant (state $iRED$), in the sense that if they are active, they cannot carry a successful packet (because l already hears i). Since R_j must not hear any other transmissions, all edges $\{(m, j) : m \in \Gamma^{-1}(j), m \neq i\}$ are blocked (state B). Furthermore, all edges $\{(m, n) : m \in \Gamma^{-1}(j), n \in \Gamma(m), n \neq j\}$ are indirectly blocked (state iB), since the blocking of (m, j) prevents m from transmitting.

The first upper bound is obtained as follows. Observe that an edge may be in several states at the same time. As illustrated in Fig. 2.20, the permitted combinations are: (B, iB), $(RED, iRED)$ and $(iB, iRED)$. However, the sets of edges

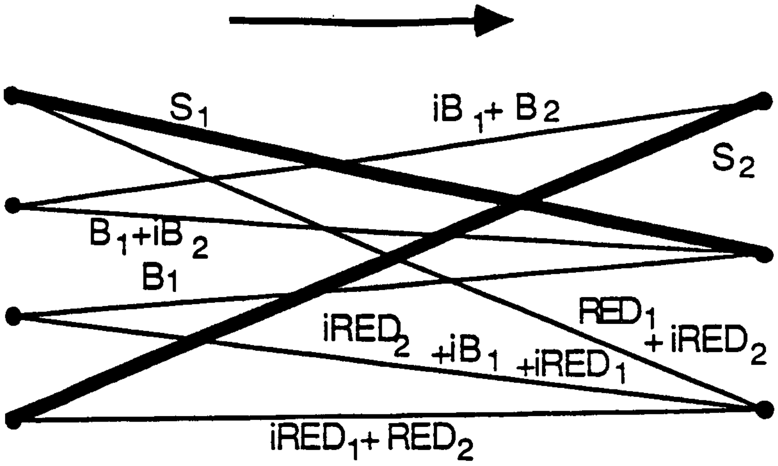


Fig. 2.20 Possible states of edges with unidirectional broadcast media.

in S, B and RED are disjoint. Furthermore, each edge in B and in RED can be associated with a single edge in S. Assuming that all the overlaps take place, (optimistic assumption,) only the edges in S, B and RED have to be counted. For each edge in S, there are $(Q-1)$ edges in RED and $(P-1)$ edges in B. Since the total number of edges in the graph is $T \cdot Q$ $(= R \cdot P = \sqrt{T \cdot Q \cdot R \cdot P})$, it follows that the total number of edges in S cannot exceed $\left\lfloor \frac{\sqrt{T \cdot Q \cdot R \cdot P}}{P+Q-1} \right\rfloor$.

The second upper bound is obtained by subtracting the minimum number of edges that are affected by a single edge that is in state S from the total number of edges. A single edge in S causes $(P-1)$ edges to be in B, and each of those causes $(Q-1)$ edges to be in iB. Similarly , it causes $(Q-1)$ edges to be in RED, and each of those causes $(P-1)$ edges to be in $iRED$. As in the previous case, an edge can be in both $iRED$ and iB, but the other combinations are impossible. Therefore, another upper bound is $\sqrt{T \cdot Q \cdot R \cdot P} - P \cdot Q + 1$.

2.6 Unidirectional, Equal-Degree $\mathcal{SBI}$'s

In this section, we consider the more general class of equal-degree $\mathcal{SBI}$'s, not necessarily with grouping or with disjoint subnetworks. Our focus will be on performance for a uniform traffic pattern, as well as on performance bounds for nonuniform patterns.

2.6.1 Maximum Throughput with Randomized Transmissions and a Uniform Traffic Pattern

Consider any unidirectional, equal-degree $\mathcal{SBI}$ with k alternate paths from each TS to each RS. Recalling that an equal-degree $\mathcal{SBI}$ corresponds to a bipartite graph with equal degrees for all transmitters and equal degrees for all receivers, proposition 2.3 can be applied to this situation. In doing so, it is also assumed that all the transmitters of a TS can operate independently, and so can the receivers of an RS. The following substitutions are made for the variables appearing in the proposition: $T = N_T \cdot C_T$; $R = N_R \cdot C_R$; $Q = \frac{k \cdot N_R}{C_T}$; $P = \frac{k \cdot N_T}{C_R}$. The result is

$$\frac{1}{e} \cdot \frac{C_T \cdot C_R}{k} \leq S_{max} \leq 0.5 \cdot \frac{C_T \cdot C_R}{k}. \tag{2.24}$$

This result is consistent with the results for a unidirectional equal-degree $\mathcal{SBI}$ with grouping and disjoint subnetworks, including the identical performance of the *MSP* and *SMP*. It should be noted that this does not contradict the difference between the bidirectional *MSP* and *SMP*. In that case, equating $\mathcal{C}_{\text{min}}$ and C does not guarantee equal numbers of intragroup paths, so the nodal degrees in the two variants may be different. It is also important to note that the class of equal-degree $\mathcal{SBI}$'s is very broad, yet the above results are the same for all. This allows the designer to incorporate other considerations into the design, without altering the performance for the uniform traffic pattern.

Let us now consider the situation wherein a TS can operate at most one of its transmitters in any given slot, and an RS can receive at most one transmission in any given slot. Each receiver is nevertheless assumed to be capable of independently deciding whether a transmission that it hears is receivable (no collision), and whether or not a receivable transmission is intended for its RS. Therefore, whenever the receivers of an RS hear at least one receivable transmission which is intended for their RS, one of those transmissions (chosen at random) is received. The transmission process of each TS is assumed to be Bernoulli ($p \cdot C_T$); the transmitter is selected at random and the destination is selected at random from among those that can hear that transmitter. To facilitate analysis, a single-path $\mathcal{SBI}$ is considered. To calculate the throughput, observe that:

(i) A receiver can hear at most one transmitter of any given transmitting station. Therefore, the reception process at a given receiver is not affected by a dependence between the transmission processes of different transmitters within the same TS. (This holds for multiple-path $\mathcal{SBI}$'s as well.)

(ii) The subsets of transmitting **stations** that can reach two receivers of the same RS are disjoint. Consequently, the packet arrival processes at two such receivers are independent. (This is not true for a multiple-path $\mathcal{SBI}$.)

From (i), it follows that the probability that a given receiver hears a receivable transmission which is intended for its RS is

$$S_R = P \cdot \frac{p \cdot C_T}{N_R}\left(1 - \frac{p \cdot C_T}{C_T}\right)^{P-1} = P \cdot \frac{p}{Q}(1-p)^{P-1} \qquad (2.25);$$

i.e., the same as in the previous case.

From (ii), it follows that the throughput of a receiving station is

$$S_{RS} = 1 - (1 - S_R)^{C_R}, \qquad (2.26)$$

and the aggregate throughput is thus

$$S = N_R \cdot S_{RS} = N_R \cdot \left[1 - \left(1 - \left(1 - P \cdot \frac{p}{Q}(1-p)^{P-1}\right)\right)^{C_R}\right] \qquad (2.27);$$

This is maximized by setting $p = \frac{1}{P}$, yielding (for $P \gg 1$)

$$S_{max} = N_R \left[1 - \left(1 - \frac{1}{e \cdot Q}\right)^{C_R}\right]. \qquad (2.28)$$

If $e \cdot Q \gg C_R$, this becomes $\frac{1}{e} \cdot C_T \cdot C_R$, which was the result in the first case. In other words, the probability of two or more receivers of the same station hearing receivable packets intended for them in the same time slot is negligible.

2.6.2 Increasing the Maximum Throughput for a Uniform Traffic Pattern by Deterministic Scheduling of Transmissions

2.6.2.1 Motivation

Having seen that a throughput of $1/e \cdot C_T \cdot C_R$ can be attained with random transmissions, it is natural to ask whether one can improve by scheduling the transmissions deterministically. Knowing that the $\frac{1}{e}$ factor represents collisions, it is clear that one should be able to achieve $1 \cdot C_T \cdot C_R$. Furthermore, for certain equal-degree $\mathcal{SBI}$'s, such as the single-path $\mathcal{SBI}$ with disjoint subnetworks and grouping (in the strong sense), this is also an upper bound. (As can be seen in Fig. 2.6, the latter can be represented by a graph with a minimum cut of $C_T \cdot C_R$, which is obviously an upper bound on concurrency.) Therefore, if it is possible to attain higher throughputs, more complicated $\mathcal{SBI}$'s must be used; also, unlike the result for randomized transmissions, which held for all equal-degree $\mathcal{SBI}$'s, the results of this section will obviously depend on the specific $\mathcal{SBI}$.

We next show $\mathcal{SBI}$'s which can achieve a higher concurrency than $C_T \cdot C_R$, while retaining the equal-degree property. Grouping is only in the weak sense, but the groups are of equal size. The description will include the logical interconnection as well as a transmission schedule. It will initially be required that C_T and C_R be even and that $C_T = C_R = C$. Some of those assumption will then be relaxed.

2.6.2.2 Achieving $1.5 \cdot C^2$ with $C_T = C_R = C$ (Even)

Logical interconnection

Divide the transmitting stations (TS) into $3C$ groups of equal size; similarly, divide the receiving stations (RS) into $3C$ groups of equal size. All stations within a group will have identical logical interconnections; we will therefore speak of the groups as if they consisted of single stations. All computations are modulo $3C$, unless stated otherwise. Since each TS has C transmitters, it follows that each transmitter must be connected to receivers of 3 groups of RS. Similarly, each receiver must be connected to transmitters of 3 groups of TS. The first transmitter of each TS in the kth group is connected to RS groups k through $k+2$; the second transmitter is connected to groups $k+3$ through $k+5$, and so on.

The exact interconnection rule, depicting the transmitter and receiver numbers in addition to the group numbers, is as follows. Let i be an odd integer, $i \in \{1, 3, ..., (C-1)\}$. The ith transmitter of a TS that belongs to the kth group is connected to the ith receiver of each RS in group $[k+(i-1)3]$ and to the $(i+1)$th receiver of the each RS in the two following groups. The $(i+1)$th transmitter is connected to the ith receiver of each RS in groups $[k+3+(i-1)3]$ and $[k+3+(i-1)3+1]$ and to the $(i+1)$th receiver of each RS in the following group. Note that for odd i, one third of the connections of an ith transmitter are to ith receivers, and the remaining two thirds are to $(i+1)$th receivers. To balance this, two thirds of the connections of an $(i+1)$th transmitter are to ith receivers and only one third are to $(i+1)$th receivers. Finally, note that the transmitter and receiver numbers can be divided into pairs $\{(1,2),\ (3,4),...,(C-1,C)\}$, and all interconnections are between numbers in the same pair. Therefore, the different pairs can be scheduled independently and concurrently without interference. The logical interconnection is valid for all values of N_T and N_R, provided that they are both divisible by $3C$. Fig. 2.21 shows the connection of a typical group.

Schedule

The schedule will be stated in terms of which (TS group, RS groups) may communicate in each time frame. Once a pair of groups is specified, the scheduling of the exact pairs of stations that may communicate can be done in many different ways, including deterministic schedules as well as any desirable access scheme. In the calculations of concurrency, we will count the number of group pairs that may communicate concurrently, thus implicitly assuming perfect utilization of each frame. Once an access scheme is specified for the scheduling within a group, the results can be multiplied by the utilization factor of that scheme. (e.g. $\frac{1}{e}$ for slotted ALOHA, 1 for TDMA.)

Since the schedules for the $\frac{C}{2}$ different pairs of transmitter numbers can be executed concurrently without conflict, the schedule will be described for a single pair; this is done by denoting the transmitter number only as “odd” or “even”. It should also be noted that the schedules for the different pairs can be executed with any desirable relative phases, and it is therefore possible to prevent frequent (and, more important, overlapping) transmissions or receptions by any given station. A schedule will be specified as a collection of triplets. The first element specifies

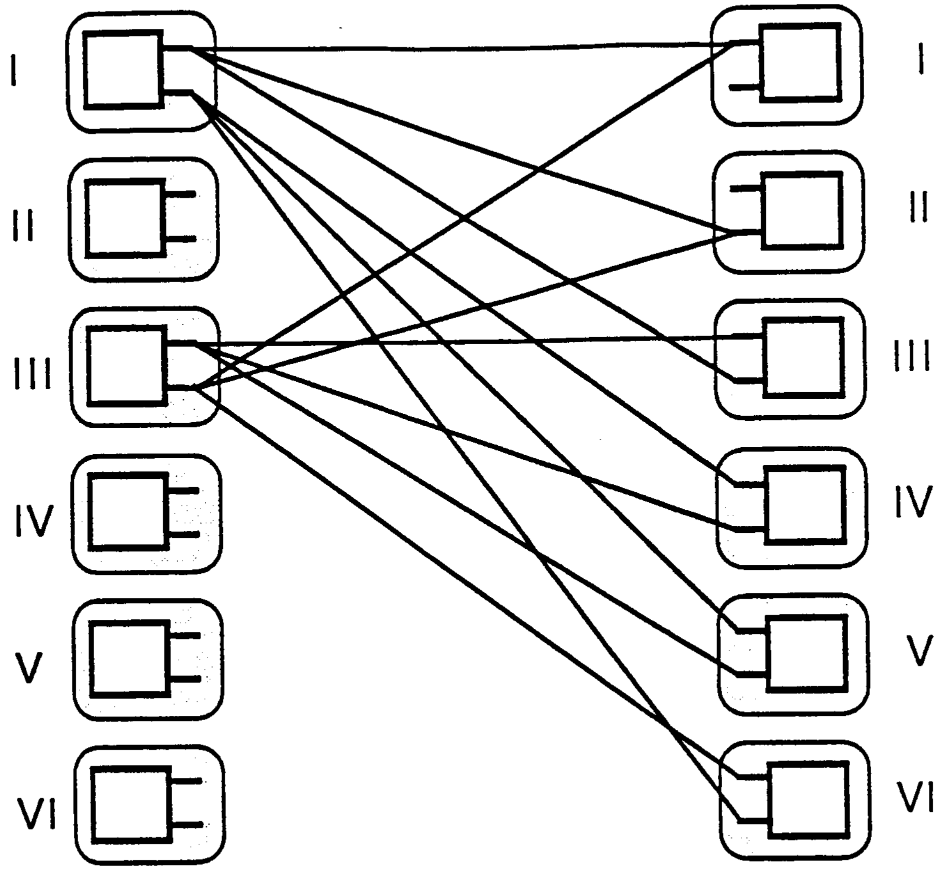


Fig. 2.21 Interconnection for $1.5C^2$ with $C_T = C_R$, both even. Each group is represented by a single station, and the interconnection is shown for TS groups I and III. ($C = 2$.)

whether the transmitter number is odd or even; the second is the number of a TS group; the third specifies which of the 3 groups to which the transmitter is connected includes the destination RS. The advantage of this description is that it is true for all transmitter numbers. The schedules will be expressed in a pseudo Pascal format. Whenever the word "concurrently" appears in a loop statement or in a begin block statement, all iterations of the loop or block are executed concurrently.

```
begin
{Mode 1. Odd-numbered transmitter to odd-numbered receiver or even to even, but not both}
 for m:=1 to 2 do
  begin
  case m of
    1: T:=odd, Connection:=1;
    2: T:=even Connection:=3;
    end
  for i:=1 to 3*C do concurrently {concurrently for all  TS groups. }
    begin
    [T, i, Connection]
    end;
  end; {for m:=1 to 2}
{Note that, by specifying T only as odd or even, it is implied that this is executed concurrently for all odd or all even transmitter numbers.}
{Mode 2. Odd-to-even and (concurrently) even-to-odd.}
for m:=1 to 4 do
  begin
  case m of
    1,3: d:=-1;
    2,4: d:=0;
    end
  case m of
    1,2: ConnectionOdd:=2, ConnectionEven:=2;
    3,4: ConnectionOdd:=3, ConnectionEven:=1;
    end
  case m of
    1,3: dgroup:=0;
    2,4: dgroup:=1;
  for i:=1 to 3*C/2 do concurrently {concurrently for all odd or all even TS group numbers. }
    begin
    j:=2*i+d;
    begin concurrently { concurrently for all odd and all even transmitter numbers.}
      [odd, j, ConnectionOdd]
      [even, j+dgroup, ConnectionEven]
      end;
    end;
  end; {for m:=1 to 4}
 {Note that this mode is executed concurrently for all transmitter numbers.}
 end.
```

The concurrency is calculated as follows. In mode 1, all $3C$ groups of TS operate concurrently, and each group employs either the odd or the even transmitter numbers; i.e., $C/2$ transmissions per group. The concurrency is therefore $1.5 \cdot C^2$. In mode 2, only every other group of TS is allowed to transmit in any given slot, but each transmitting group may use all transmitter numbers, so again we have $1.5 \cdot C^2$.

2.6.2.3 Achieving $1.5 \cdot C_T \cdot C_R$ with $C_T \neq C_R$ (Even)

Logical interconnection

Again, one third of the connections of an odd-numbered transmitter will be to odd-numbered receivers and the remaining two thirds will be to even-numbered receivers. The opposite holds for even-numbered transmitters. As before, the first transmitter of a TS in group k will be connected to receivers of RS's in $\frac{1}{C_T}$ of the groups, beginning with group k; the second transmitter will be connected to receivers of RS's in the next batch of groups, etc. The main difference is in the fact that each transmitter of any given station is now connected to receivers with all numbers. The fact that the number of connections of an odd-numbered transmitter to an even-numbered receiver is twice as large as the number of connections of such a transmitter to odd-numbered receiver, combined with the fact that the unit of resolution is a group, forces us to have $\frac{3}{2}C_T C_R$ groups of TS and the same number of RS groups.

The exact interconnection rule is as follows (All math is modulo the number of groups.) Let i be an odd integer, $i \in \{1, 3, 5, ..., C_T - 1\}$. The ith transmitter of a TS in group k is connected to the first receiver of each station in RS group $[k + (i-1) \cdot \frac{3}{2}C_R]$, to the second receiver of each RS in the two following groups, to the third receiver of each RS in group $[k + (i-1) \cdot \frac{3}{2}C_R + 3]$, to the fourth one of each RS in the two following groups, and so on up to and including the C_Rth receivers of group $[k + i \cdot \frac{3}{2}C_R]$. Fig. 2.22 illustrates the interconnection for a typical TS group. It should be noted that this interconnection can also be used when $C_T = C_R$, but it requires an unnecessarily large number of groups.

Schedule

For brevity, the schedule will only be outlined. The details can be reconstructed using the previous examples along with the specification of the logical interconnection.

Mode 1. Odd-numbered transmitter to odd-numbered receiver or even to even, but not both.

For all groups of TS (concurrently), let an ith transmitter, (i odd,) transmit to one of its odd-numbered destination receivers. Repeat as necessary to cover all such destinations and all odd is. Then, do the same using even-numbered transmitters and their even-numbered destination receivers.

Mode 2. Odd to even **and** even to odd.

For all odd-numbered groups of TS (concurrently), let an ith transmitter, i odd, transmit to one of its even-numbered destination receivers. At the same time, for all odd-numbered groups of TS, let an $(i + 1)$th transmitter (even) transmit to one of its odd-numbered destination receivers. By examining the interconnection, it can be seen that this combination can reach only one half of the possible destinations. (It will be repeated as necessary to achieve that.) To complete the schedule, it will be repeated in a similar way for even-numbered groups and then for even-numbered groups doing the even-to-odd and odd-numbered groups doing odd-to-even and vice versa. Also, each portion of the schedule will be repeated for all values of i.

The concurrency is calculated as follows. In mode 1, each group of TS transmits once in each slot, and this is also the concurrency. The number of groups is $1.5 \cdot C_T \cdot C_R$. In mode 2, the number of concurrent transmissions is again equal to the

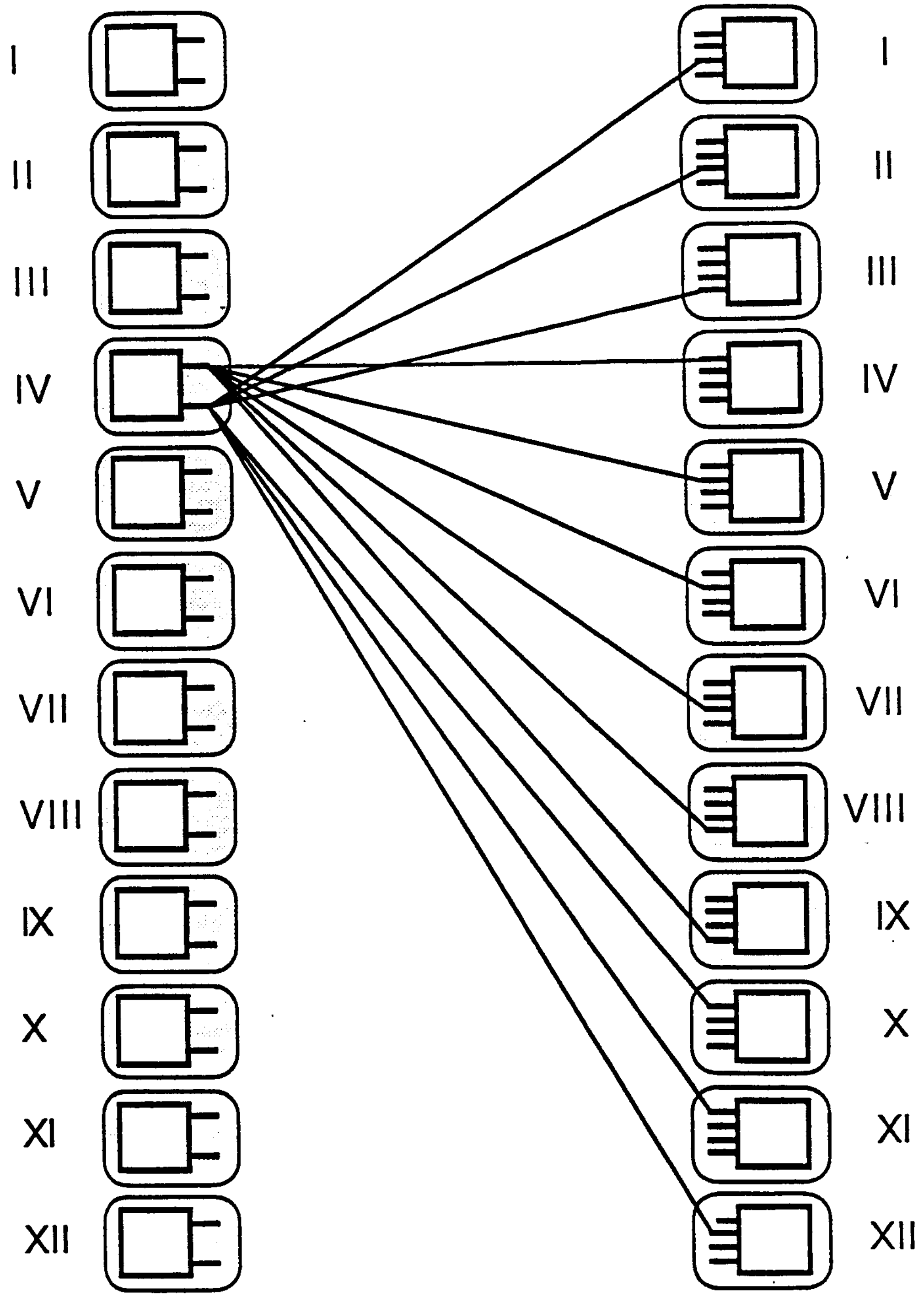


Fig. 2.22 Interconnection for $1.5C_TC_R$ with $C_T \neq C_R$, but both even. Each group is represented by a single station, and the interconnection is shown for a single group of TS's. ($C_T = 2$, $C_R = 4$.)

number of groups.

2.6.2.4 Allowing Odd C_R

The above method can be extended to the case of odd C_R. However, the resulting interconnection is not equal-degree; odd-numbered transmitters reach one fewer group than do even-numbered ones. Nevertheless, recalling that the number of groups is $1.5 \cdot C_T \cdot C_R$, the relative difference in degree is negligible. The interconnection rule is essentially the same as in the even case, and is illustrated in Fig. 2.23 for a typical group. Mode 1 of the schedule is the same as before, except that the odd-to-odd part will be repeated more times. Mode 2 is also the same, except that the even-to-odd portion must be executed more times than the odd-to-even. In those cycles, the concurrency is only $0.75 \cdot C_T \cdot C_R$.

To calculate the average concurrency, let us determine the number of slots necessary to complete the schedule. (We assume that each group consists of a single station, since the number of stations will not affect the concurrency.) The total number of connections is the square of the number of groups; i.e., $(1.5 \cdot C_T \cdot C_R)^2$. Most of those connections are carried out at a rate of $1.5 \cdot C_T \cdot C_R$. However, for each group, transmissions of each even-numbered transmitter to any of its two last (odd-numbered) destination groups cannot be matched by odd-to-even transmissions. These connections are therefore carried out at the rate of $0.75 \cdot C_T \cdot C_R$. (Although an even-numbered transmitter has only one more destination group than an odd-numbered one, recall that the odd-to-odd schedule was repeated more times than the even-to-even.) The total number of such connections is $1.5 \cdot C_T \cdot C_R \cdot \frac{C_T}{2} \cdot 2$. The average concurrency is therefore

$$1.5 \cdot C_T \cdot C_R \cdot \frac{1.5 \cdot C_R}{1 + 1.5 \cdot C_R}. \tag{2.29}$$

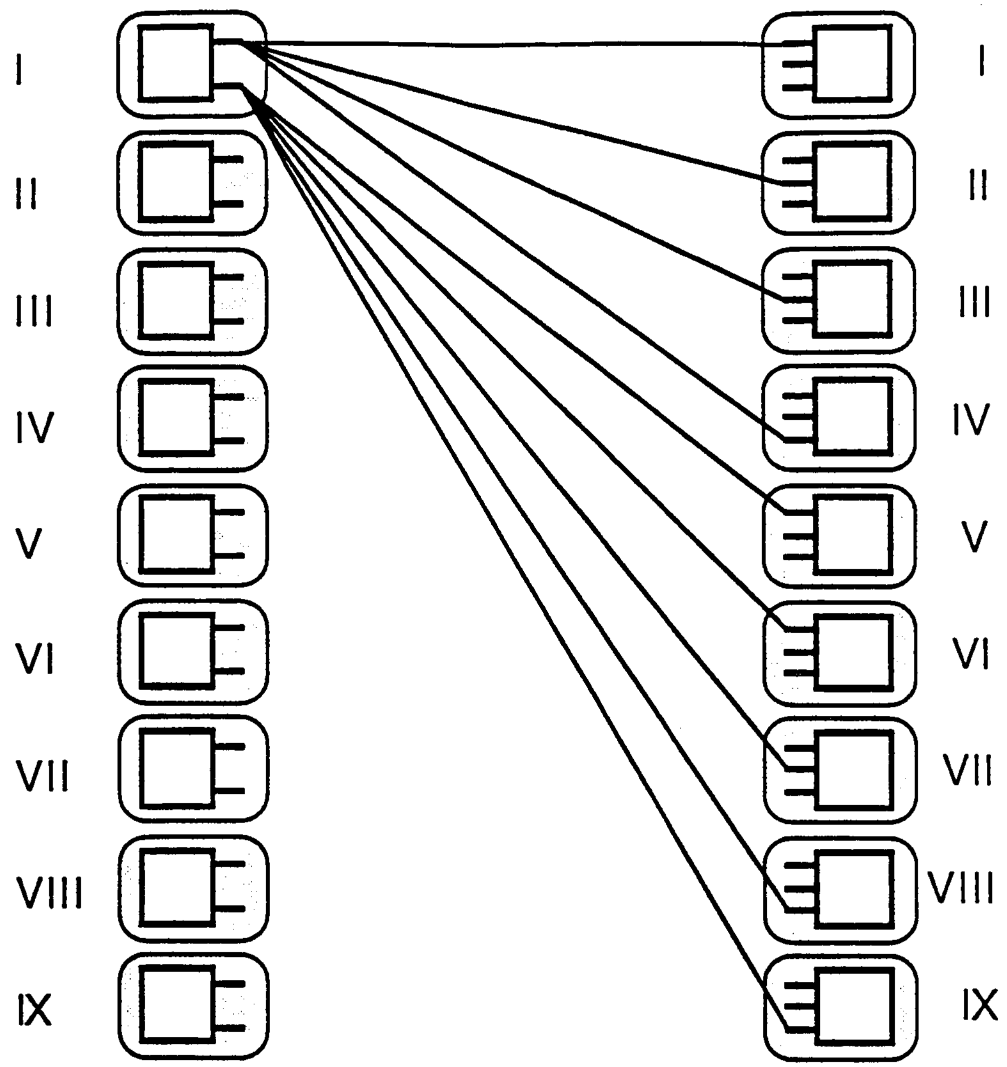


Fig. 2.23 Interconnection for $1.3C_TC_R$ with $C_T \neq C_R$; C_T is even, but C_R is odd. Each group is represented by a single station, and the interconnection is shown for a single group of TS's. ($C_T = 2,\ C_R = 3$.)

As expected, the concurrency approaches $1.5 \cdot C_T \cdot C_R$ when C_R is large.

2.6.2.5 Beyond $1.5 \cdot C^2$

Let us return to the case of $C_T = C_R = C$, for some even C. We now show how to achieve a concurrency of $1.81C^2$.

Logical interconnection

The idea is identical to that used earlier, except that the stations are now divided into $6 \cdot C$ groups. Fig. 2.24 depicts the interconnection for two consecutive groups. The concurrency in mode 1 is $2 \cdot C^2$, but in mode 2 it is only $\frac{2}{7} \cdot 6C \cdot C = 1.71 \cdot C^2$. Recalling that two thirds of the slots are spent in mode 2 and only one third in mode 1, the average concurrency is $1.81 \cdot C^2$. The exact schedule is omitted due to the complexity of stating it. However, it again involves only pairs of transmitter and receiver numbers and is constructed by educated selections of source and destination groups in each sub-mode. This is a dense interleaving of sources and destinations, which avoids collisions of transmissions at their destinations while permitting them to overlap at other RS groups.

We have not been able to obtain a theoretical upper bound on the maximum concurrency for a uniform traffic pattern.

2.6.3 Maximum Concurrency of Equal-Degree $\mathcal{SBI}$'s

Consider the class of k-path, unidirectional equal-degree $\mathcal{SBI}$'s, with N_T transmitting stations, each equipped with C_T transmitters and N_R receiving stations, each equipped with C_R receivers. We wish to establish a tight upper bound on the maximum "peak" concurrency that can be provided by such $\mathcal{SBI}$'s. Contrary to the previous sections, in which we considered uniform traffic patterns, the question here can be stated as follows: what is the maximal length of a source-destination list, such that there is an $\mathcal{SBI}$ in the above class which can accommodate the entire list concurrently? Since equal-degree $\mathcal{SBI}$'s correspond to equal-degree bipartite graph, we apply proposition 3, making the following substitutions: $Q = \frac{k \cdot N_R}{C_T}$ and $P = \frac{k \cdot N_T}{C_R}$. The resulting upper bound is

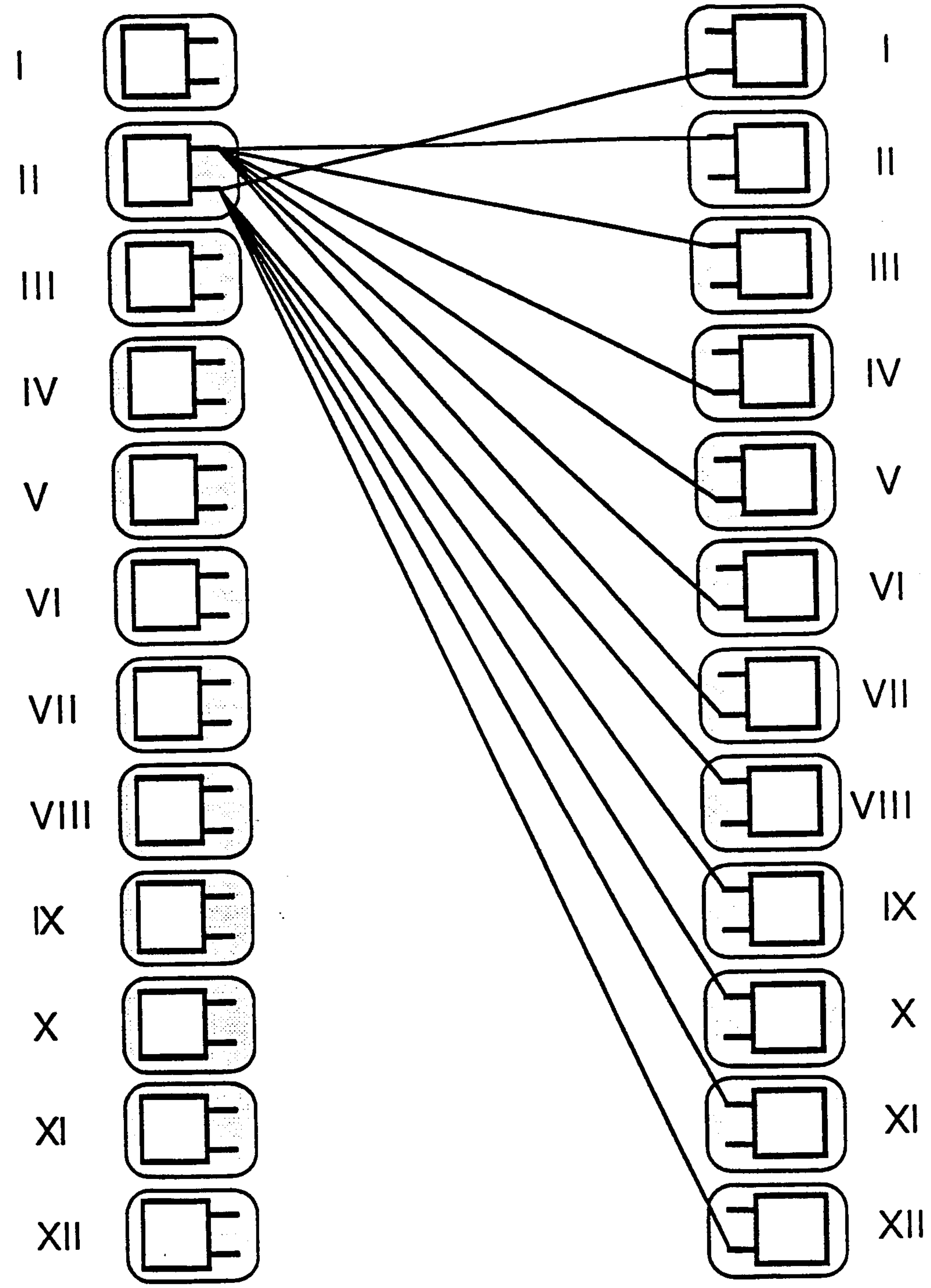


Fig. 2.24 Interconnection for $1.8C^2$ with $C_T = C_R$, even. Each group is represented by a single station, and the interconnection is shown for a single TS group. ($C = 2$.)

$$S^* \leq \mathcal{C}_{\min} \left\{ \left\lfloor \frac{N_T \cdot N_R}{\frac{N_T}{C_R} + \frac{N_R}{C_T} - 1/k} \right\rfloor , \; k \cdot N_T \cdot N_R \cdot \left(1 - \frac{k}{C_T \cdot C_R}\right) + 1 \right\} . \qquad (2.30)$$

It should be noted that $k \leq C_T, C_R$. The 2nd term in the bound comes into play only when $C_T = C_R = 1$. To get a feeling for the size of this bound, note that when $N_T = N_R = N$, $C_T = C_R = C$, and $k = 1$, the bound is very close to $0.5 \cdot N \cdot C$, which is one half of the total number of single-destination packets that can ever be transmitted concurrently, since a transmitter can have at most one ongoing transmission.

This upper bound cannot be achieved with any equal-usage $\mathcal{SBI}$. For example, any $\mathcal{SBI}$ which also has disjoint subnetworks can carry at most $\frac{C_T \cdot C_R}{k}$ concurrent transmissions.

When one is given a list of source-destination pairs that are to be accommodated concurrently, (the length of the list may not exceed the upper bound,) it is not always possible to design satisfactory equal-usage $\mathcal{SBI}$. Some of the necessary conditions for success are:

- No TS or RS may appear in the list more than C_T and C_R times, respectively.
- If TS i and RS j appear on the list C_T and C_R times, respectively, then (i, j) must be on the list. (Otherwise, there is no way of connecting i to j without causing interference between two of the paths on the list.)
- If RS j appears on the list m times, then at least $(P-1) \cdot m$ TS's whose connection to j is not on the list must appear on the list fewer than C_T times.
- Additional arguments can be formulated along the same lines, but their complexity grows very rapidly.

There are cases in which the upper bound can be reached; those are characterized by a uniform distribution of the connections. For example, in the case that

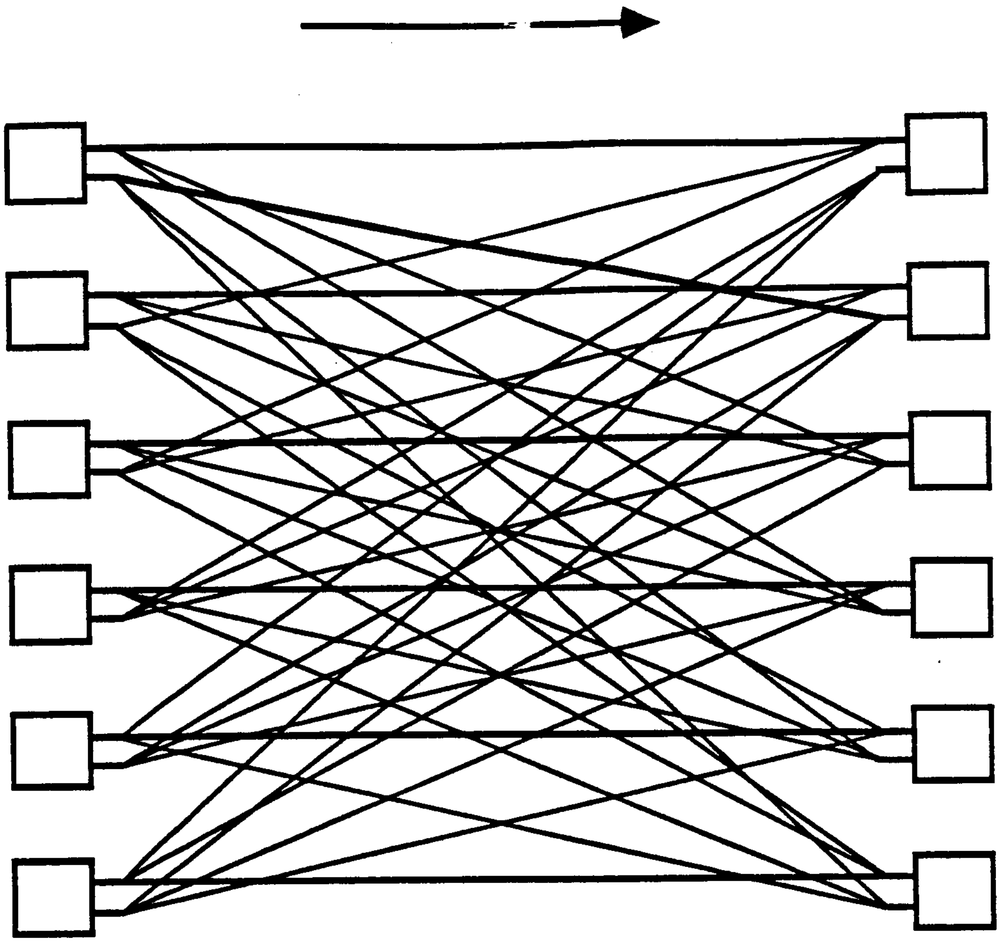


Fig. 2.25 An equal-degree $\mathcal{SBI}$ which achieves the upper bound on concurrency. The thick edges are mutually noninterfering. $N = 6$; $C = 2$; maximum concurrency= 7.

$N_T = N_R = N$, $C_T = C_R = C$, $k = 1$, the list may consist of connections between TS i to RS's $i+1$ through $i + \lfloor C/2 \rfloor - 1$ (modulo N), for all $1 \leq i \leq N$. In Fig. 2.25 we show such an example. Since there are cases in which the upper bound can be achieved, we consider it to be tight.

The maximum destination-independent throughput is the same, since one is free to choose the sources and destinations when constructing the list for which the maximum concurrency is obtained. The maximum reception rate is $N \cdot C$, which is the total number of receivers, and can be obtained with any $\mathcal{SBI}$ in which there is a subset of transmitters that exactly covers all receivers. (e.g. one transmitter in each group in an $\mathcal{SBI}$ with grouping and disjoint subnetworks.)

2.7 Accommodation of Nonuniform Traffic Patterns

2.7.1 Possible Approaches

Nonuniform traffic patterns can be accommodated in any of the following ways or a combination thereof:

(i) Designing an $\mathcal{SBI}$ whose worst case performance exceeds the required one.

(ii) Tailoring the $\mathcal{SBI}$ to the specific traffic pattern.

(iii) Given a traffic pattern and an $\mathcal{SBI}$, assigning transmitters and receivers to its input and output ports, respectively, so as to uniformize the load.

The first approach was discussed in an earlier section, in the context of designing an $\mathcal{SBI}$ according to performance tradeoffs. In this section, it will be shown that similar results can be obtained by optimizing the design for a uniform traffic pattern, and operating the $\mathcal{SBI}$ with some randomization, thereby limiting the performance-degradation due to nonuniformity of traffic pattern. Using a pattern-independent $\mathcal{SBI}$ has the advantage of simplicity and flexibility. For example, no changes have to be made when the pattern changes. However, the cost is relatively high. The second approach, which represents the other extreme, has the drawback of possibly complicated construction and is very inflexible, (has to be completely redesigned whenever the traffic pattern changes), but can attain the highest performance for a given number of transmitters and receivers. This approach will be touched upon briefly. The third approach is a compromise. Although the interconnection graph must be tuned to the traffic pattern, this tuning is restricted to the renumbering of source and destination nodes; (i.e., transmitters and receivers). Viewing the passive interconnection as a black box with input and output ports, the black box is thus unchanged, and the tuning to a given traffic pattern is achieved by assigning transmitters and receivers to input and output ports, respectively. The application of this approach to single-path $\mathcal{SBI}$'s with disjoint subnetworks is the focus of this section; it will be broadened to include certain modifications to the passive interconnection.

2.7.2 Two-Hop Transmissions with Randomization on an Equal-Degree, Single-Path, Unidirectional $\mathcal{SBI}$ with Disjoint Subnetworks and Grouping

In section 2.3.5, concurrency with a uniform traffic pattern was traded for worst-case concurrency through the design of the $\mathcal{SBI}$. In this section, a similar tradeoff will be achieved through the operation of an $\mathcal{SBI}$ which is optimized for a uniform traffic pattern, such as the single-path, equal-degree $\mathcal{SBI}$ with disjoint subnetworks and grouping. TS's and RS's are assumed to be paired to form bidirectional stations. The idea, which was originally proposed by Valiant and Brebner [30] for the Hypercube interconnection, is as follows. Instead of transmitting directly to the destination, the source station transmits a message to a randomly chosen station. That station, in turn, forwards the message to its true destination.

In our case, there is clearly no sense in sending a message to a randomly chosen station in the group of the destination. Therefore, if some station from the destination group is selected as the recipient of the first hop transmission, it is replaced with the true destination; the latter, of course, will not bother to forward the message. Similarly, the source should never choose a recipient from among the stations in its own group, unless the destination is in that group.

To illustrate the effect of this scheme, let us consider the worst case, wherein all traffic is between two groups. In this case, one transmission per slot goes directly from the source to the destination; the remaining ones go to a randomly chosen station which is in a different group than the source as well as the destination. (There are $C-2$ such groups.) Consequently, the subnetworks used for the second hop are different from those used for the first hop, and the process is pipelined. The source may thus transmit, and the destination may receive, $C-1$ transmissions per slot. The concurrency with a uniform traffic pattern drops to $C^2/2$; although all subnetworks are utilized in each slot, each transmission occupies two slots. (The concurrency is actually slightly higher, since $1/C$ of the transmission go only one hop. On the other hand, the actual throughput will be lower by a factor of e due to the inefficiency of the access scheme that is implicit in this algorithm.) Finally, it is worth noting that increasing the number of hops cannot increase the guaranteed concurrency, since the number of alternate paths into or out of any given group is C, which is thus an obvious upper bound on guaranteed concurrency.

The above is an extreme example. To achieve intermediate results, the source would be required to transmit directly to the destination with probability p, and to use the above algorithm with probability $(1-p)$. Although this scheme is far superior to achieving the tradeoff through the design of the $\mathcal{SBI}$, it should be noted that two-hop communication can be viewed as a violation of the single-hop connectivity. If nodes are permitted to forward traffic, it is not clear why the topology should provide single-hop connectivity. If this restriction is lifted, there are interconnections that can outperform $\mathcal{SBI}$ in a very substantial way. For example, if stations are placed on a grid, and are then interconnected by "row" buses and "column" buses, each station needs only 2 transceivers; yet, for a uniform traffic pattern, the concurrency is $\sqrt{N}$, and at most two hops are required.

2.7.3 Custom $\mathcal{SBI}$'s

Customization of an interconnection can be done to various extents and under various constraints, such as standard stations, $\mathcal{SBI}$'s with disjoint subnetworks, etc. This discussion will be restricted to a presentation of several ideas and guidelines.

One very important observation in an $\mathcal{SBI}$ is that the actual bandwidth available to a station is not simply the transmission rate times the number of transmitters and receivers; to obtain the actual bandwidth, one must multiply the above by the fraction of time in which each transmitter and receiver may be utilized. Therefore, when the traffic pattern indicates that a certain station must carry a large fraction of the traffic, one should consider designing the $\mathcal{SBI}$ so that that station's transmitters receive a large fraction of the time on the channels over which they transmit.

Another important observation is that, whenever there is a subset of stations such that the traffic pattern representing communication among them is uniform, they should best be interconnected by a single-path, equal degree $\mathcal{SBI}$.

With unidirectional media, it is possible to construct $\mathcal{SBI}$'s which can attain a concurrency of $N \cdot C/2$ for certain traffic patterns while retaining the equal-degree property and hence the C^2 concurrency for a uniform traffic pattern. Therefore, if at all possible, one should consider using unidirectional media.

Since $\mathcal{SBI}$'s with disjoint subnetworks have many practical advantages, such as the fact that they can be implemented with unidirectional as well as bidirectional media, the customization of such $\mathcal{SBI}$'s warrants special attention. One approach is to assume that a certain throughput, (say in units of messages per slot,) can be attained. The traffic pattern matrix can then be multiplied by that constant, thereby becoming the actual traffic matrix. To design the custom $\mathcal{SBI}$, certain steps should be taken. First, check to see whether the traffic matrix has any entries which are close to 1.0 or perhaps exceed that. Such entries should best be accommodated by dedicated point-to-point links, since the hardware utilization is maximized. Next, try to identify clusters of stations, such that the pattern of traffic among them is nearly uniform, and construct a single-path $\mathcal{SBI}$ for each such cluster. The number of transmitters and receivers per station in that $\mathcal{SBI}$ will be determined by the traffic it must handle. Whenever a subnetwork is constructed and some of the traffic is assigned to it, that traffic must obviously be subtracted from the traffic matrix, and the appropriate transmitters and receivers marked as used. Due to the discrete nature of the assignment, there may often remain excess capacity in a subnetwork. In such an event, additional stations may be interconnected. Those stations will be such that the traffic between them and at least some of the stations that already belong to the subnetwork is nonzero. Finally, it is always possible to use one transmitter and one receiver per station and construct a broadcast bus, thus guaranteeing that there is a transmission path between any pair of stations.

A configuration which is quite typical of many networks consists of several large hosts, each with a set of users. For simplicity, let us also assume that the sets of users are identical, and that all nonzero matrix entries are user-host and are equal to each other. The initial tendency is to let each host be served exclusively by several subnetworks; each user then has one transmitter and receiver per host, and is connected to each host through one of the host's subnetworks. However, for a given total number of transmitters and receivers, the maximum throughput will be attained by the allocation which maximizes the number of subnetworks, and this may differ from the above. To illustrate this, consider the case of U users, H hosts, and a total of $C \cdot U$ transceivers. If k transceivers are allocated to each host, each user is left with $C - k \cdot H/U$. Since each user must communicate with H hosts using its limited number of transceivers, each subnetwork must have at least $H/(C - k \cdot H/U)$ hosts as its members. The resulting number of subnetworks is then

$$\text{Number of subnetworks} = \frac{k \cdot H}{\frac{H}{C-\frac{k \cdot H}{U}}} = k \cdot \left(C - \frac{k \cdot H}{U}\right). \tag{2.31}$$

Maximizing with respect to k yields

$$k = \frac{C}{2} \cdot \frac{U}{H}; \tag{2.32}$$

i.e., $C/2$ transceivers per user, and the remaining ones go to the hosts. There are

$$\frac{C^2}{4} \cdot \frac{U}{H} \tag{2.33}$$

subnetworks, each of which interconnects $\frac{2H}{C}$ hosts with the same number of users. This is clearly different from the straightforward approach. (For simplicity, we used transceivers rather than separate transmitters and receivers.)

Having seen some general ideas for customization of $\mathcal{SBI}$'s, let us now turn to the main issue in this section, namely the assignment of transmitters and receivers to ports of "standard" $\mathcal{SBI}$'s.

2.7.4 $\mathcal{SBI}$ with Disjoint Subnetworks and Customized Assignment of Transmitters and Receivers to Subnetworks

We start out by considering the single-path, equal-degree $\mathcal{SBI}$ with disjoint subnetworks and grouping. The problem of finding an optimal assignment of stations to groups will be shown to be NP-Complete in the strong sense. As a result, there cannot be any pseudo-polynomial algorithms for its solution, even if one restricts the number of bits representing the entries of the traffic matrix. Nevertheless, a heuristic approach will be outlined; this approach is likely to achieve good results in many practical situations. Next, the requirement of equal degree will be relaxed;

C - 2

i.e., groups of different sizes will be permitted. We will show that, although the problem remains very hard, a simplified version, in which the TS assignment and that of RS are done independently, can guarantee a concurrency of C. (Provided that it is feasible.) Furthermore, this version lends itself to efficient approximation algorithms. Next, the grouping requirement will be relaxed for either the TS's or the RS's; this will permit the simplified version to achieve substantially better results. Finally, a monte carlo method, known as "simulated annealing", will be described and its adaptation to the assignment problems will be outlined.

2.7.4.1 Single-Path, Equal-Degree $\mathcal{SBI}$ with Grouping (Groups of equal sizes)

Optimal assignment of stations to groups

Given a single-path $\mathcal{SBI}$ with equal group sizes, along with a traffic matrix, we wish to assign stations to groups so as to maximize throughput. Throughput is maximized by assigning the stations to groups so as to uniformly distribute the load among the subnetworks. This problem will be proven to be NP-complete in the strong sense, by reducing to it a problem known as "3-partition", which is defined as follows.[29]

Instance. A finite set $\mathcal{A}$ of $3m$ elements, a bound $B \in \mathcal{Z}^+$, and a size $s(a) \in \mathcal{Z}^+$ for each $a \in \mathcal{A}$, such that each $s(a)$ satisfies $B/4 < s(a) < B/2$ and such that $\sum_{a \in \mathcal{A}} s(a) = m \cdot B$.

Question. Can $\mathcal{A}$ be partitioned into m disjoint sets $\mathcal{S}_1, \mathcal{S}_2, ..., \mathcal{S}_m$ such that, for $1 \leq i \leq m$, $\sum_{a \in \mathcal{S}_i} s(a) = B$? (Notice that the above constraints on the item size imply that every such $\mathcal{S}_i$ must contain exactly 3 elements from $\mathcal{A}$.)

Proposition 2.4. Optimal assignment of stations to groups of equal size is NP-Complete in the strong sense.

Proof. it is clearly in NP, since generating an assignment and evaluating the resulting load on each subnetworks can both be done in polynomial time. By trying out all possible assignments and comparing them, the optimum can be found.
Reduction from 3-partition: Given an instance of 3-partition, construct an instance of our problem as follows. Let $C_R = m$ and $N_T = 3 \cdot m$; then, let all elements in the ith row of the traffic matrix equal $s(i)$. Clearly, the assignment of RS to groups is not important; each group of TS will consist of 3 stations, and therefore uniform loading will be achieved if and only if the instance of 3-partition has an optimal solution.

Heuristic, suboptimal assignment of stations to groups

In a typical network, one can identify large "hosts", small users of those hosts, and small independent stations, such as workstations. Each host has a set of users, and there can also be clusters of workstations which communicate with each other. (Intracluster.) Such subsets will be referred to as "functional subsets." In order to equally utilize all transmitters and receivers of a given host, it is desirable to spread its users uniformly among the different groups. Similarly, the workstations of any given cluster should also be spread among the groups. Finally, it is also desirable to uniformize the load on all subnetworks.

The heuristic approach which is outlined here is based on the fact that the traffic matrix is often sparse, and can be broken down into submatrices which correspond to typical scenarios. The idea is to identify "functional" subsets of stations; the stations in each such subset should be spread uniformly among the groups. Since "functional" subsets may overlap, "atomic" subsets are constructed. An atomic subset is either contained in a given functional subset or disjoint from it, and the collection of atomic subsets constitutes a partition of the set of stations. Clearly, if the load presented by each atomic subset is uniformly distributed among the groups, the same will be true for the functional subsets. The major steps are as follows:

Identification of the "functional" subsets:

1) Identify the hosts. If their identities are not obvious, sum up each row and each column of the traffic matrix and look at the sums. Hosts will have sums that far exceed those of other stations. The set of hosts is a functional subset.

2) For each host, identify its users. These will be the dominant entries in the row and column representing the host. The set of users of each host is a functional subset.

3) Similarly, identify workstation clusters. Each such cluster is a functional subset.

Construction of the "atomic" subsets and assignment to groups:

4) Construct the collection of atomic subsets.

Note: Thus far, a station has been treated as a single entity. The following steps, however, must be conducted separately for TS's and RS's.

5) Spread the hosts among the groups so as to uniformize the groups' loads. Since the number of hosts is usually not very large, one can either try out all possibilities or use some simple heuristic. Note that the hosts constitute a functional subset. This subset is given special treatment since its distribution has a major effect on the success of the load uniformization. In fact, since the number of hosts is small, one need not worry about the requirement for equal group sizes. Any skew that results from this stage will be compensated for in the next one.

6) Distribute the members of each non-host atomic subset among the groups. In this phase, attention must be paid to the equal group size requirement; nevertheless, the real goal is achieving a uniform distribution of the load. Therefore, sort the members by their traffic in descending order, and then distribute them among the groups in a round robin fashion. Since the number of elements in an atomic subset is generally not an integer multiple of the number of groups,

the round for a given subset should commence at the group following the one at which the round for the previous subset ended. Furthermore, it would be wise to start with the more dominant atomic subsets, and attempt to order the subsets so that consecutive atomic subsets have at least one common functional superset.

7) Make small corrections, such as balancing the number of stations in each group.

As has already been stated, this is not an optimal algorithm. There may even be pathological cases in which it performs very poorly. However, in many common situations, such as 5-20 hosts and several hundred small users, it can provide a realistic way of getting high performance from the interconnection.

There are cases in which traffic can not be uniformized. An example is a case wherein one host's traffic constitutes more than $1/C$ of the total traffic. Since that host is connected to exactly C subnetworks, at least one of those must carry more than $1/C^2$ of the total traffic, which contradicts uniform loading of the subnetworks. This problem can be solved by equipping such a host with additional transmitters and receivers and making it a member of several groups; viewed differently, such a host is represented by several stations, each of which belongs to a different group. Strictly speaking, this is a not a single-path interconnection, but the violation applies only to a very limited number of stations (at most C).

2.7.4.2 Single-Path $\mathcal{SBI}$ with Grouping (Not equal-degree; groups of unequal sizes)

At the outset, it should be noted that, depending on the physical medium, changing the size of a group may require an internal change to the $\mathcal{SBI}$; strictly speaking, this is a deviation from the constraint that the customization be limited to the assignment of transmitters and receivers to ports. However, in many environments, such as a broadband LAN, the physical connection of transmitters and receivers to the interconnection is identical, and the assignment to groups amounts to a change in frequency, which is performed at the transmitter or receiver. In such environments, it is sensible to include the case of groups of unequal size in the category now being discussed.

Optimal assignment of stations to groups of unequal sizes

Proposition 2.5. Optimal assignment to groups of not necessarily equal size is NP-complete.

Proof. Clearly, it is in NP. We now reduce the knapsack problem to it.

Knapsack. Given a set $\mathcal{A}$ of m elements, with a size $s(a) \in \mathcal{Z}^+$ for each $a \in \mathcal{A}$, and a constant $B \in \mathcal{Z}^+$, determine whether there is a subset of elements whose sum is B.

Given an instance of knapsack, an instance of our problem is created as follows: $C = 2$; the traffic matrix is an $(m+1) \times (m+1)$ matrix. The entries in any given row are all identical. Each of the first m rows corresponds to one of the m elements in the set $\mathcal{A}$. The entries in each of those rows are equal to the size of the corresponding element. The elements in the remaining row are all equal to $|2B - \sum_{a \in \mathcal{A}} s(a)|$. (The assignment problem is a unidimensional one, since all columns of the matrix are identical). To show that the solutions to the two problems are equivalent, two cases must be considered: (i) $2B \geq \sum_{a \in \mathcal{A}} s(a)$, and (ii) $2B < \sum_{a \in \mathcal{A}} s(a)$. If the answer to the knapsack is "yes", there will also be a perfect assignment: in case (i), one group will consist of the solution to knapsack, and the other group will consist of the remaining elements (including the additional one). In case (ii), one group will consist of the solution to knapsack plus the additional element, and the other will have the remaining elements. The converse is also true: in case (i), the solution to knapsack is the group that does not include the additional element; in case (ii), it is the group that includes the additional element, less that element.

The above proposition showed that finding an optimal assignment of TS's to groups as well as that of RS's to groups are both NP-Complete. Unfortunately, the combined assignment is even more complicated. This is primarily due to the fact that a distribution of the TS's among the groups so that all TS groups generate equal amounts of traffic, combined with a distribution of the RS's among the groups so that all RS groups have to sink equal amounts of traffic, does not guarantee that all subnetworks are equally loaded. On the positive side, independent optimal assignments of TS's and RS's to groups do guarantee a concurrency of at least C. (If at all feasible.) This is explained as follows: in the worst case, all the traffic of each TS group will be intended for a single RS group; this, however, will still permit C subnetworks to be utilized. The guaranteed concurrency of C justifies a search for approximate solutions to the "independent" assignment problem.

Approximation algorithm for assigning elements to groups of unequal sizes

A known NP-Complete problem that bears a close resemblance to the problem of assigning stations to groups is the *bin packing* problem [29]. There, a set of objects is given, each with some nonnegative size; the objects are to be placed into bins of equal, known capacities. The objective is to accommodate all objects in the minimal number of bins. Using bin terminology, the assignment problem is to place objects into a given number of bins of equal sizes in a way that minimizes the bin size. For the bin-packing problem, there are very simple algorithms that are guaranteed to come quite close to the optimum. We next describe one such algorithm, and show how to transform our problem to a bin packing problem.

The algorithm, referred to as FFD, (first fit decreasing,) is as follows: initially, sort the objects by size in decreasing order. Then, place each object in the first bin than can accommodate it. The worst case performance of this is [29]

$$FFD(I) = \frac{11}{9} \cdot OPT(I) + 4, \tag{2.34}$$

where I is an instance of bin-packing and OPT stands for optimum. Clearly, this approximation improves as the number of bins grows.

To transform our assignment problem to bin packing, choose a bin size and solve bin-packing. If the number of bins is larger than C, increase the bin size and repeat. If it is less than or equal to C, decrease the bin size and repeat. Continue until the change in bin size is of little significance. Clearly, the number of steps is logarithmic in the required resolution and hence poses no complexity problem.

A similar algorithm , which appears to be a direct translation of FFD to our problem, but whose performance we have not analyzed, is as follows: sort the objects by size in decreasing order. Then, assign each object to the bin with the least content. (Here, there are C bins of unlimited size, and the goal is to minimize the maximum content.)

2.7.4.3 Single-Path $\mathcal{SBI}$ with Grouping of TS or RS But Not Both (Groups of unequal sizes; not equal-degree)

From a computational complexity point of view, it is desirable to have the assignment of TS's independent from that of RS's. However, such an independence can cost a factor of C in performance. We now show how relaxing the grouping requirement on TS's or on RS's, but not on both, can greatly improve the situation.

Consider a situation in which the TS's are grouped such that all groups generate equal amounts of traffic. From the grouping of TS's and the constraints of disjoint subnetworks and single-path $\mathcal{SBT}$'s, it follows that each RS must use one of its C receivers to listen to each of the TS groups. Since it obviously does not matter which receiver listens to which group, let us assume that the ith receiver listens to the ith TS group. Once the assignment of TS's to groups has been completed, the amount of traffic that is destined for each individual receiver is also known.

Since each TS group can transmit over C subnetworks, the remaining assignment problem is as follows: for each i, $1 \leq i \leq C$, distribute the N_R ith receivers among the C subnetworks which belong to the ith TS group, so as to uniformize the load on those subnetworks.

Although this approach still cannot *guarantee* good results, it can fail only when More than $1/C$ of the traffic of a given TS group is destined to a single *receiving station*. This is far less likely to occur than the condition in the previous case, namely that more than $1/C$ of the traffic from a given TS group be destined for a given RS *group*. Furthermore, in the event that there is a problem, it should be very easy to identify it and make a manual correction, since very few stations would be involved. Needless to say, each of the assignment phases can be performed using the bin-packing approach.

2.7.4.4 Simulated Annealing

Simulated annealing [31] is a heuristic monte-carlo method for finding "good" solutions to complicated problems. Although simulated annealing is a suboptimal method, and is not *guaranteed* to do well, it has been shown to be a very good practical method for problems in which the step and the cost computation are both very simple, and in which the problem is likely to have many "good" solutions. For example, El Gamal and Sperling have applied it to the generation of "good" codes [32] for data compression. The general idea of the algorithm is to start with a feasible solution, and then make incremental, highly randomized changes in order to obtain "neighboring" solutions. A new solution replaces the current one if it is better; if it is worse, it may still be accepted with some nonzero probability. This directs the search in a manner that improves the solution, yet prevents locking into local minima. The algorithm proceeds as follows:

Initially:

1) Define a cost measure over the set of feasible solutions. (The lower the cost, the better the solution.) In our case, the cost can be the maximum load over all subnetworks.

2) Define an annealing step. This is the method of selecting the next candidate feasible solution. The step must not be deterministic. In the case of groups of equal size, a possible step is to choose two TS or two RS at random (members of different groups), and swap them.

3) Choose some initial feasible solution and compute its cost.

4) Select an initial "temperature". (Will be explained shortly.)

Repeat:

5) Take an annealing step. Accept the new solution with probability

$$P = \min\left\{ e^{\frac{\text{old cost} - \text{new cost}}{T}}, 1 \right\}, \tag{2.35}$$

where T is a "temperature". In other words, a better solution is always accepted; a worse one is accepted with a probability that is small if the solution is much worse but can be high if it is marginally worse. Also, the probability of acceptance of a worse solution is higher at higher temperature. This is repeated a "sufficient" number of times.

6) Lower the temperature and return to step 5). This is repeated until the temperature is sufficiently low and the algorithm converges to a local minimum.

At extreme temperatures, simulated annealing reduces to well known approaches: at infinite temperature, it is simply a random search; at zero temperature, it is a greedy algorithm that finds a local minimum.

The initial temperature is usually set to be equal to the cost of the worst solution, and the final temperature is lower than the optimal solution. In our case, the initial temperature can be set equal to the total traffic, and the final one to $\frac{\text{total traffic}}{100 \cdot C^2}$. The exact settings as well as the time spent at each temperature and the increment by which the temperature is reduced are determined from experience.

Simulated annealing is equally applicable to groups of unequal size. The step will be choosing one TS or one RS, and a group at random, and assigning the chosen station to the chosen group. It is also possible to use simulated annealing in conjunction with heuristic approaches. For example, fix the assignment of the hosts and apply simulated annealing to the remaining stations. The method can even be used to perform specific steps in the heuristic, such as the assignment of the stations in a specific atomic subset.

2.8 Delay Performance of a Single-Path $\mathcal{SBI}$

2.8.1 Outline

So far, concurrency served as the primary measure of performance. To give a more complete picture, we now compare the performance of the single-path, equal degree, unidirectional $\mathcal{SBI}$ with disjoint subnetworks and grouping with those of parallel buses ($\mathcal{PBI}$) and the single broadcast bus ($\mathcal{SBB}$). (Delay is the time interval from the creation of a message until its successful delivery to its destination).

Since our topic is interconnection design, we wish to consider only topology-dependent issues, thus excluding ones that depend on the channel access scheme. (The latter will receive some treatment in the next chapter.) We will therefore assume that messages that need to be transmitted are queued, and are transmitted as soon as the channel becomes available when they are at the head of the appropriate queue. This is equivalent to assuming a perfect access scheme. To further facilitate the comparison, message generation will be assumed to follow a Poisson process, and message lengths will be exponentially distributed. This will permit the use of simple queueing models for which analytical results are available. The same ideas can be adapted to other situations. The comparison will be based on a uniform traffic pattern. Whenever that is not the case, the performance of $\mathcal{SBI}$ can simply be determined based on the most heavily loaded subnetwork.

2.8.2 Queueing Models

$\mathcal{SBB}$ will be modeled as an $M/M/1$ queue. The single-path $\mathcal{SBI}$ will be modeled as a collection of independent $M/M/1$ queues. The relative arrival rate to each of those queues is a function of the traffic pattern; nevertheless, they can be analyzed separately. $\mathcal{PBI}$ will be assumed to permit only bit-serial transmissions; however, concurrent transmissions between a pair of stations are possible. Therefore, it will be modeled as an $M/M/m$ queue; a single job can be served by only one server, but this can be any of the servers.

Let λ and $\frac{1}{\mu}$ denote the mean rate of packet generation and the mean packet transmission time, respectively. For an $M/M/1$ queue, the mean packet delay is then given by

$$T = \frac{\frac{1}{\mu}}{1 - \frac{\lambda}{\mu}}. \tag{2.36}$$

For an $M/M/m$ queue, let p_0 denote the probability that the system is empty and $\rho \triangleq \frac{\lambda}{m \cdot \mu}$. It has been shown that [33]

$$p_0 = \left[\sum_{k=0}^{m-1} \frac{(m \cdot \rho)^k}{k!} + \left(\frac{(m\rho)^m}{m!} \right) \left(\frac{1}{1-\rho} \right) \right]^{-1}. \tag{2.37}$$

It can also be shown, by finding the mean number of messages in the system and using Little's theorem, that the mean delay is given by

$$T = \frac{p_0}{\lambda} \cdot \left\{ \sum_{k=1}^{m} \frac{k \cdot (m\rho)^k}{k!} + \frac{m^m}{m!} \cdot \frac{\rho^{m+1}}{(1-\rho)^2} \cdot (m \cdot (1-\rho) + 1) \right\}. \tag{2.38}$$

2.8.3 Comparison

The delay performance will be conducted for the same three cases that were used in the throughput comparison. For clarity, λ_0 and μ_0 will denote the values of λ and μ for $\mathcal{SBB}$, and the values for $\mathcal{SBI}$ and $\mathcal{PBI}$ will be expressed in terms of these. Also, m will denote the number of buses in $\mathcal{PBI}$; the number of subnetworks in the $\mathcal{SBI}$ will be a case-dependent function of m. As before, S represents the aggregate throughput, B is the transmission rate, and C is the number of transmitters and receivers per station.

In all three cases, average delay for $\mathcal{SBB}$ is

$$T_{\mathcal{SBB}} = \frac{\frac{1}{\mu_0}}{1 - \frac{\lambda_0}{\mu_0}}. \tag{2.39}$$

1) Equal S for all three schemes; equal B for $\mathcal{SBI}$ and $\mathcal{PBI}$.

$\mathcal{PBI}$ with m buses: the following substitutions should be made in the $M/M/m$ results.

$$\lambda = \lambda_0; \; \mu = \frac{\mu_0}{m}; \; \rho = \frac{\lambda_0}{\mu_0}. \tag{2.40}$$

$\mathcal{SBI}$ with m subnetworks: substituting

$$\lambda = \frac{\lambda_0}{m}; \; \mu = \frac{\mu_0}{m},$$

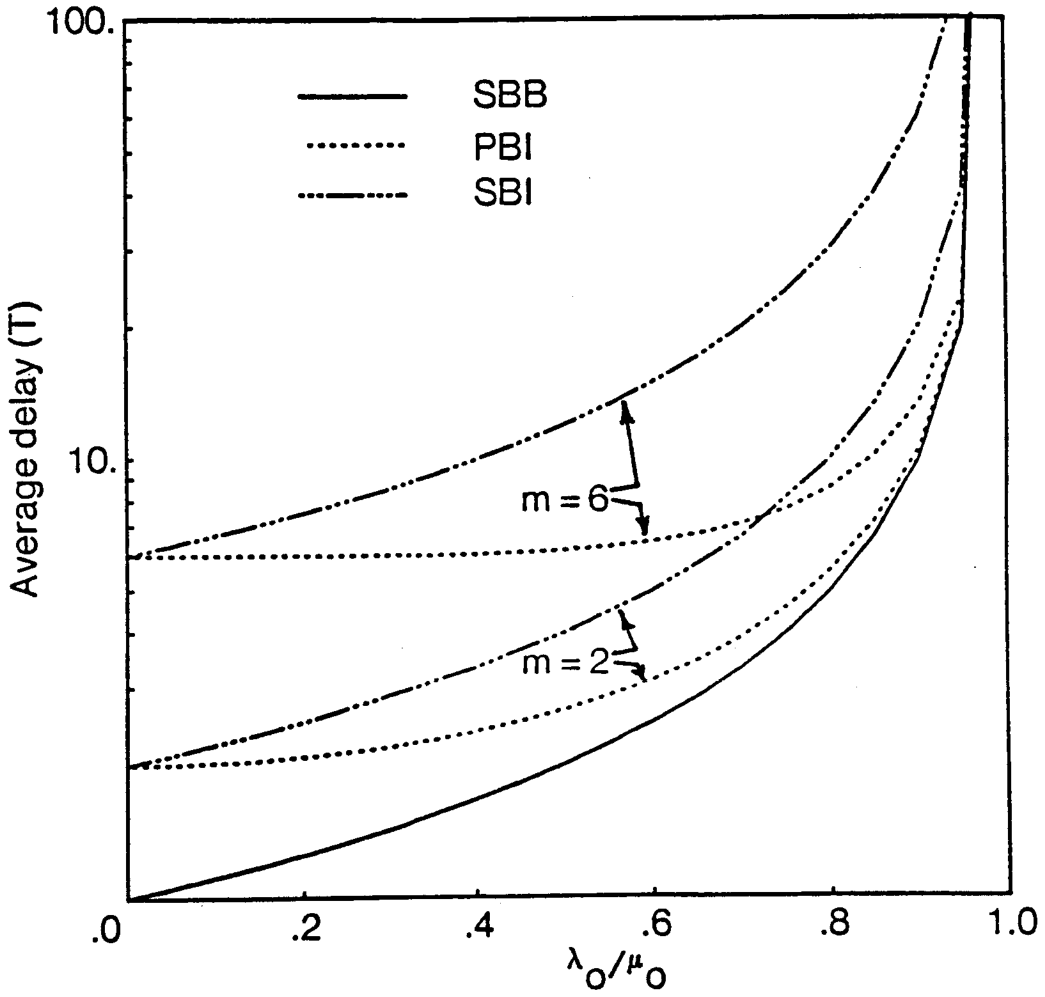


Fig. 2.26 Delay comparison of $\mathcal{SBI}$, $\mathcal{PBI}$ and $\mathcal{SBB}$. Equal S for all; equal B for $\mathcal{SBI}$ and $\mathcal{PBI}$.

in the $M/M/1$ results yields

$$T_{\mathcal{SBI}} = m \cdot T_{\mathcal{SBB}}. \tag{2.41}$$

Fig. 2.26 presents the mean packet delay as a function of λ_0/μ_0 for $\mathcal{SBB}$, $\mathcal{PBI}$ and $\mathcal{SBI}$; results are presented for $m = 2, 6$. At low values of λ_0/μ_0, $\mathcal{PBI}$ and $\mathcal{SBI}$

perform equally well, but both are m times worse than $\mathcal{SBB}$. This is so because in this range delay consists primarily of the transmission time, which is m times longer for $\mathcal{PBI}$ and $\mathcal{SBI}$ than for $\mathcal{SBB}$ due to the m-fold fragmentation of the aggregate bandwidth. As λ_0/μ_0 increases, the queueing delay becomes a major factor. Here, the fact that $\mathcal{PBI}$ permits any message to be transmitted over any channel enhances its performance with respect to $\mathcal{SBI}$ and brings it closer to that of $\mathcal{SBB}$. As the load on the interconnection grows even further, the delay with $\mathcal{PBI}$ becomes essentially equal to that with $\mathcal{SBB}$, whereas that with $\mathcal{SBI}$ remains m times higher.

2) Equal S for all schemes; equal C for $\mathcal{SBI}$ and $\mathcal{PBI}$.

$\mathcal{PBI}$: same as 1). (m buses.)

$\mathcal{SBI}$: there are now m^2 subnetworks, each with a transmission rate that is $1/m^2$ of the aggregate bandwidth. Therefore,

$$T_{\mathcal{SBI}} = m^2 \cdot T_{\mathcal{SBB}}. \tag{2.42}$$

Fig. 2.27 presents the mean packet delay as a function of λ_0/μ_0 for $\mathcal{SBB}$, $\mathcal{PBI}$ and $\mathcal{SBI}$; results are presented for $m = 2, 6$. The performance of $\mathcal{SBB}$ as well as that of $\mathcal{SBI}$ are the same as before. As for $\mathcal{SBI}$, the aggregate bandwidth is now fragmented into m^2 channels, so the delay is m^2 times higher than that with $\mathcal{SBB}$.

3) Equal $B \cdot C$ for all schemes; equal B and C for $\mathcal{PBI}$ and $\mathcal{SBI}$.

$\mathcal{PBI}$: same as 1). (m buses.)

$\mathcal{SBI}$: There are m^2 subnetworks, each with transmission rate μ/m. Therefore, $\lambda = \lambda_0/m^2$, $\mu = \frac{\mu_0}{m}$, and

$$T_{\mathcal{SBI}} = \frac{\frac{m}{\mu_0}}{1 - \frac{\lambda_0}{m \cdot \mu_0}}. \tag{2.43}$$

Fig. 2.28 presents the mean packet delay as a function of λ_0/μ_0 for $\mathcal{SBB}$, $\mathcal{PBI}$ and $\mathcal{SBI}$; results are presented for $m = 2, 6$. The performance of $\mathcal{SBB}$ as well as that

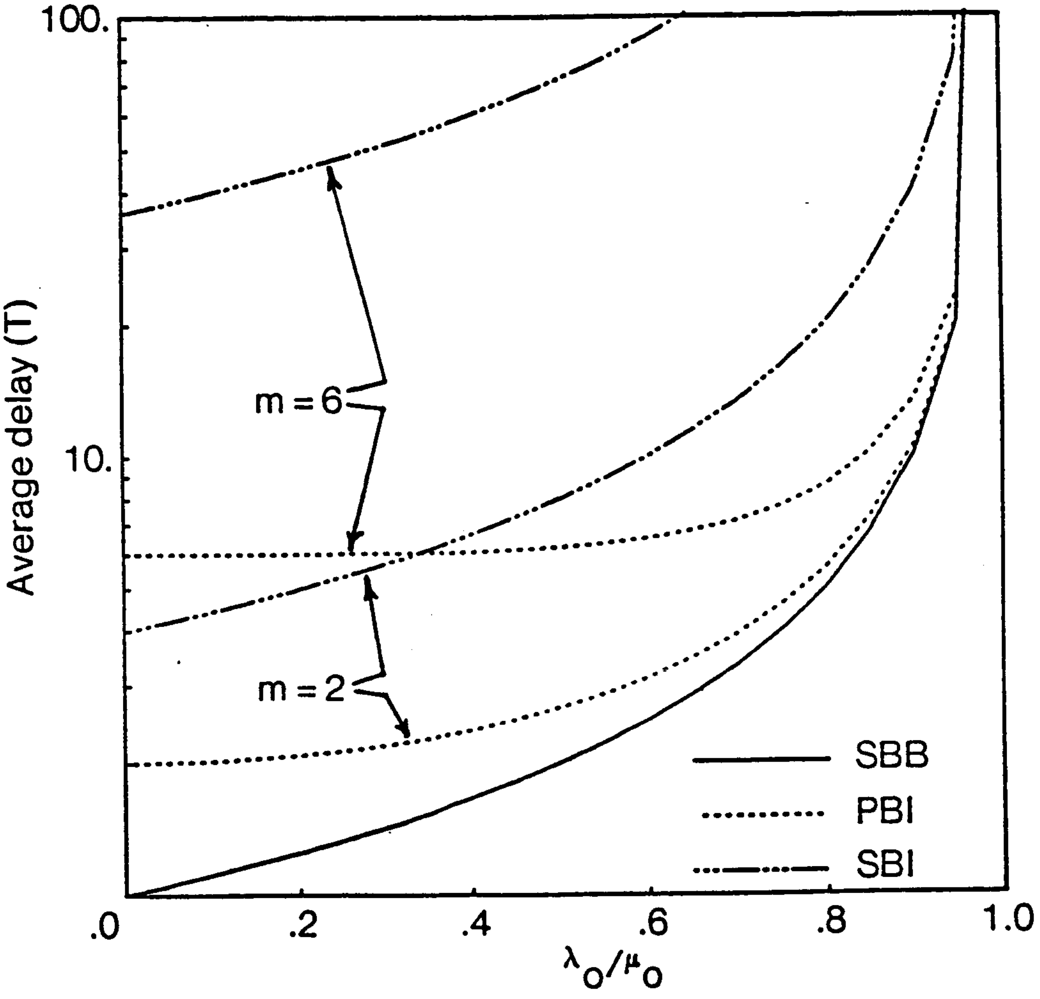


Fig. 2.27 Delay comparison of $\mathcal{SBI}$, $\mathcal{PBI}$ and $\mathcal{SBB}$. Equal S for all; equal C for $\mathcal{SBI}$ and $\mathcal{PBI}$.

of $\mathcal{SBI}$ are again unchanged. As for $\mathcal{SBI}$, there are m^2 subnetworks, (as in case 2,) but each of them has the same bandwidth as the m buses of $\mathcal{PBI}$. (Recall that the constraint here is equal transmission rate per station, not equal bandwidth of the medium.) As a result, the delay at low load is equal to that with $\mathcal{PBI}$ and

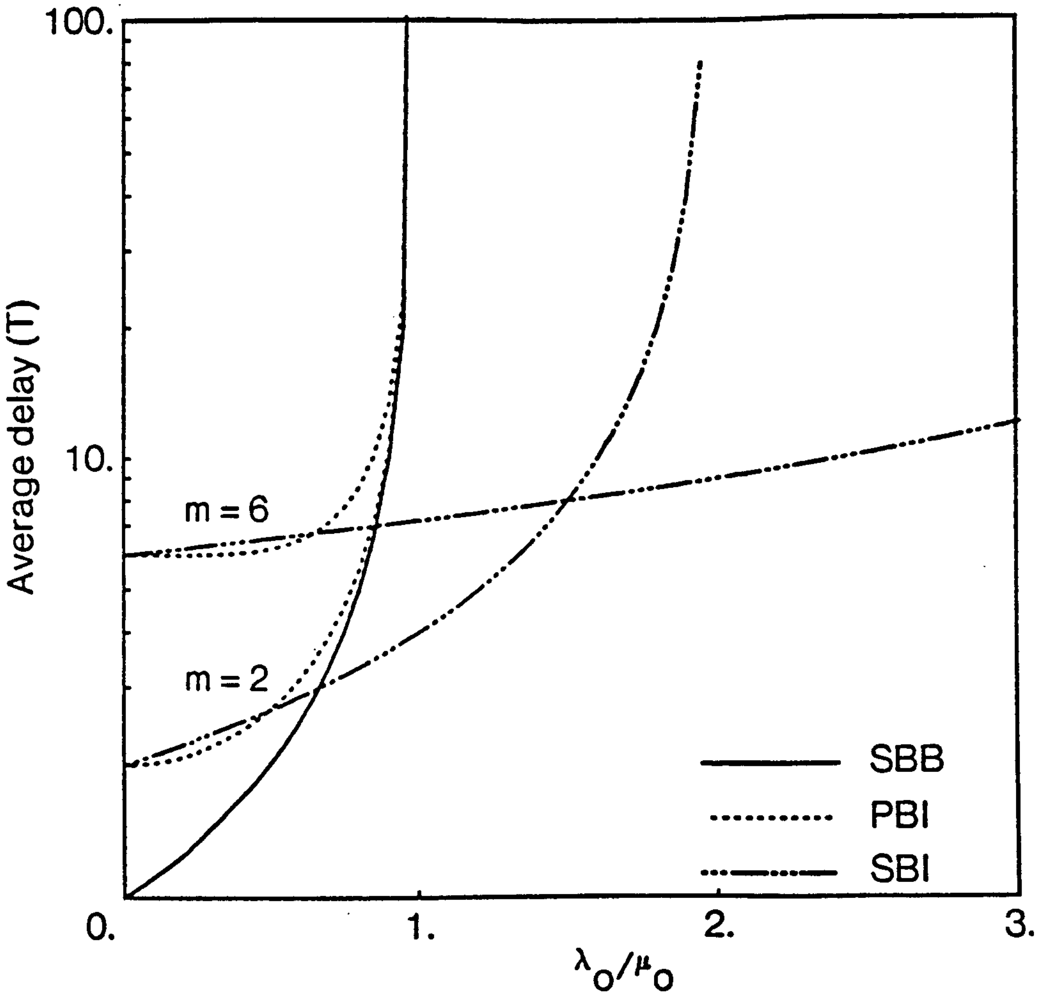


Fig. 2.28 Delay comparison of $\mathcal{SBI}$, $\mathcal{PBI}$ and $\mathcal{SBB}$. Equal S for $\mathcal{SBB}$ and $\mathcal{PBI}$; equal B and C for $\mathcal{SBI}$ and $\mathcal{PBI}$.

m times higher than with $\mathcal{SBB}$. As the load increases, $\mathcal{PBI}$ becomes superior to $\mathcal{SBI}$ due to the pooling of the servers; however, the difference is very slight. As the load increases further, the fact that the capacity with $\mathcal{SBI}$ is m times higher begins to play a major roll, and $\mathcal{SBI}$ begins to outperform $\mathcal{PBI}$. Finally, as the load approaches $\lambda_0/\mu_0 = 1$, the delay with $\mathcal{SBB}$ and $\mathcal{PBI}$ grows very rapidly, whereas that with $\mathcal{SBI}$ exhibits only a moderate growth, since the load on an $\mathcal{SBI}$ subnetwork is only $\frac{\lambda_0}{m^2}/\frac{\mu_0}{m} = \frac{1}{m} \cdot \frac{\lambda_0}{\mu_0}$. In fact, $\mathcal{SBI}$ outperforms $\mathcal{SBB}$ for

$$\frac{\lambda_0}{\mu_0} > \frac{m-1}{m-\frac{1}{m}}. \tag{2.44}$$

This is of course in addition to the fact that $\mathcal{SBI}$ can carry m times more traffic with finite delay.

Summary

Under the constraint of a fixed aggregate transmission bandwidth, relative delay is determined primarily by the degree to which the bandwidth is fragmented. (The higher the degree of fragmentation, the larger the delay.) For equal degrees of fragmentation, $\mathcal{PBI}$ outperforms $\mathcal{SBI}$ due to the pooling of the servers. As far as delay is concerned, it is therefore desirable that the hardware savings of $\mathcal{SBI}$ (compared with $\mathcal{PBI}$) be in the form of a reduced number of transceivers rather than a reduced transmission rate.

Under a constraint of fixed bandwidth per station, the comparison between $\mathcal{SBB}$ and $\mathcal{PBI}$ is not affected. However, the performance of $\mathcal{SBI}$ improves due to the fact that the aggregate transmission bandwidth increases as the fragmentation increases. This counteracts the negative effects of fragmentation, and $\mathcal{SBI}$ can actually outperform $\mathcal{SBB}$ and $\mathcal{PBI}$.

2.9 Summary

This chapter was devoted to the study of single-hop interconnections. When stations are equipped with a single transmitter and a single receiver, the only such interconnection is a single broadcast bus. However, equipping stations with multiple transmitters and receivers was shown to create a rather rich design space. This space becomes even richer if one uses unidirectional broadcast media. The class of selective-broadcast interconnections was defined and several attributes were suggested for the characterization of $\mathcal{SBI}$'s. Various performance results were then obtained for some of the subclasses.

Mutual noninterference between paths was defined, and was shown to be the correct generalization of path-disjointness so as to include broadcast media. Mutual-noninterference between two paths is the necessary and sufficient condition for successful coexistence of transmissions over them. Properties of interference were stated and proved, and it was used in computing the concurrency of various $\mathcal{SBI}$'s.

For a uniform traffic pattern, selective-broadcast interconnections were shown to be superior to $\mathcal{PBI}$ in terms of throughput for a given amount of hardware at the stations, (equal C and B,) achieving a throughput that grows quadratically with the amount of hardware. Furthermore, the average delay was also shown to be smaller in this case. The performance of certain $\mathcal{SBI}$'s was shown to be sensitive to the traffic pattern, and various ways of overcoming this were proposed. In fact, if the traffic pattern is known in advance, the performance of an appropriately constructed $\mathcal{SBI}$ can grow at least quadratically with the amount of hardware per station. If unidirectional media are used and overlapping subnetworks are allowed, the concurrency can come close to 50% of the theoretical maximum; i.e., one half of the transmitters can transmit concurrently.

Throughout this chapter, unidirectional $\mathcal{SBI}$'s were thought of as connecting a set of transmitting stations to a set of receiving stations. However, they can also be thought of as providing bidirectional connectivity between two groups of stations, but not among stations within each group. This is useful when a group of users is to be connected to a group of servers, but neither inter-user nor inter-server communication is required. Another very important application is memory-processor interconnection, in which all inter-processor communication is via shared memory and there is no inter-memory communication. For such a use to be practical, the medium must be able to support communication in both directions with no interaction between them. A good example is optical fibers. Note that this is not the same as a bidirectional medium; the difference is in the star couplers. In a bidirectional medium, a signal entering a coupler at one of its ports comes out on all ports; in a unidirectional medium, a coupler's ports are divided into two groups, (earlier referred to as input and output,) and a signal entering the coupler through a port that belongs to one group exits only over all ports of the other group.

$\mathcal{SBI}$'s are single-hop interconnections, and it was therefore natural to concentrate on their use for single-hop communication. It is, nevertheless, interesting to note that an $\mathcal{SBI}$ can also be used to simulate regular multi-hop interconnections, such as grids and hypercube. To see this, let us again think of the unidirectional $\mathcal{SBI}$ as operating in only one direction, and let each TS be paired with an RS to form a bidirectional station. Recalling the tight upper bound of slightly more than $N \cdot C/2$ on the concurrency of an equal-degree $\mathcal{SBI}$, it follows that a regular interconnection with a nodal degree k can be simulated (in one step) by an equal-degree $\mathcal{SBI}$ with $C = 2k$. This simulation is not efficient in terms of wiring cost or power budget, but the single-hop connectivity, which is obviously maintained, along with the C^2 concurrency in the single-hop mode, may offer interesting advantages in the multi-processor domain.

The main conclusion to be drawn from this chapter is that fragmentation of the stations' resources can result in an increase in network throughput which is proportional to the degree of fragmentation. Furthermore, the only penalty in delay is in terms of the transmission time, and is only relevant at low loads.

Fault-tolerance was not discussed separately, since the number of transmitters, receivers or links that can be removed without breaking the single-hop connectivity is equal to the guaranteed concurrency $\mathcal{C}_{\text{min}}$. If multi-hop transmissions are permitted in the presence of failures, the number of tolerable faults in the single-path $\mathcal{SBI}$ becomes C.

In this dissertation, $\mathcal{SBI}$'s are being discussed primarily in the context of computer networks. In fact, the next chapter is devoted to issues pertaining to the implementation and operation of an $\mathcal{SBI}$ in this domain. It is nevertheless important to note that the concept of $\mathcal{SBI}$ is much more general, and can be applied to a variety of domains. We conclude this chapter with two examples of the applicability of $\mathcal{SBI}$'s to different domains.

Multi-hub express mail. Currently, express mail carriers operate in the following manner. At the end of each day, they send as many airplanes as necessary from each city they serve to a central location, called a hub. The packages are sorted at the hub, and are then placed on airplanes that deliver them to the destination cities. The use of a single hub has the advantage of flexibility, as well as some other economic advantages. However, there are also several drawbacks, such as congestion at the hub airport and the fact that packages often fly much longer distances than necessary. One might therefore want to use multiple hubs, while guaranteeing no more than one stop-over on the way from any source to any destination, and restricting the number of planes that may take off or land at each city to C. A single-path $\mathcal{SBI}$ with disjoint subnetworks and grouping is the scheme that maximizes the number of hubs under these constraints. In this application, hubs correspond to subnetworks, flights correspond to transmitters and receivers, and sources and destinations correspond to TS's and RS's, respectively.

Inter-language translation. Consider a company that offers *direct* translation services among a set of N languages. The company employs M translators, each of whom is taught at most L languages and is capable of translating between any two of them. Several interesting questions may be posed; for example, what is the minimum number of translators required to provide translation capability among the N languages, subject to the constraint that, for each pair of languages, there are at least k translators who can translate between them?

Let us convert the terminology of this domain to $\mathcal{SBI}$ terminology as follows: languages correspond to stations; translators correspond to subnetworks; individual transceivers have no direct parallel, but the number of transceivers of the ith station, C_i, is equal to the number of translators that learn the ith language; the number of languages that a translator learns corresponds to the number of members of the corresponding subnetwork. Using this terminology conversion, the correspondence to an $\mathcal{SBI}$ is shown as follows: the direct-translation requirement, i.e., no intermediate languages, corresponds to a single-hop communication requirement; the requirement for translation capability between any two languages corresponds to single-hop connectivity, thereby turning the interconnection into an $\mathcal{SBI}$; finally, the requirement that any given translator be able to translate in both directions between any two languages he knows makes the interconnection a bidirectional $\mathcal{SBI}$.

Having shown the correspondence with $\mathcal{SBI}$, the question that was asked in the translation domain is equivalent to the design of an equal-degree, bidirectional (k, C)-path $\mathcal{SBI}$ subject to an upper limit on the number of stations per subnetwork. All the results of the section that dealt with bidirectional $\mathcal{SBI}$'s can be applied here.

Chapter 3

Implementation, Operation and Applicability of $\mathcal{SBI}$'s

3.1 Overview

The previous chapter was devoted to a theoretical study of selective-broadcast interconnections, which were treated as logical interconnections, represented by directed graphs. The number of transmitters and receivers per station and their transmission rates were assumed to be the only contributors to the cost of implementation, and the access schemes used for sharing channels were assumed to be ideal. The current chapter is devoted to the presentation of more practical issues, such as methods for achieving separation between subnetworks, operation of $\mathcal{SBI}$'s in conjunction with various access schemes and the effect of the access scheme on its performance relative to that of $\mathcal{PBI}$. Special attention will be given to issues pertaining to the fiber optic implementation of $\mathcal{SBI}$'s, such as power budget and the number of stations that can be accommodated, as well as to other practical considerations. The applicability of $\mathcal{SBI}$'s will also be discussed. Throughout the chapter, the discussion will focus on the unidirectional, equal-degree, single-path $\mathcal{SBI}$ with grouping and disjoint subnetworks; this $\mathcal{SBI}$ can be implemented using unidirectional as well as bidirectional media. Occasional comments will address other $\mathcal{SBI}$'s.

3.2 Separation of the Subnetworks, and Hardware-Savings

The figures presented in the previous chapter, such as Fig. 2.6, suggest a spatial separation between the subnetworks, and call for C_T transmitters and C_R receivers per station. Nevertheless, separation can also be achieved in the frequency domain, polarization, angle [34], (when relevant,) and others, and the actual number of transmitters per station can sometimes be as low as one. In this section, two forms of separation which are non trivial will be discussed. For convenience, fiber-optic terminology will be used, but the ideas are equally applicable to other implementations. Also, the possibilities and implications of saving transmitters and receivers will be explained.

3.2.1 Separation in the Time Domain

A particularly intriguing and perhaps somewhat confusing form of separation is in the time domain. The idea is to divide the time axis into segments of C^2 slots, and let the ith slot in each segment correspond to the ith subnetwork. The slot length is slightly larger than the transmission time of a packet, and each slot is shared among the transmitting members of the corresponding subnetwork using some access scheme. Since a single channel is being used, there is obviously no benefit over a single bus in terms of concurrency. The benefit is in the fact that if each station is equipped with C transmitters and receivers, any given transmitter or receiver must be ready for use during at most $\frac{1}{C^2}$ of the time, and the station as a whole needs to be potentially engaged in communication during only $\frac{1}{C}$ of the time. Furthermore, the "active" slots for any given transmitter or receiver are known and equally spaced. (With a single bus, a receiver must be ready at all times; with $\mathcal{PBI}$, it would have to be ready $1/C$ of the time.) This greatly facilitates the design of the buffers and other interface mechanisms of the transmitters and receivers, since they are now guaranteed to have ample time to prepare new packets for transmission or to pass received packets on to the hosts.

Such an implementation could even employ a single physical transmitter and a single physical receiver per station. However, transmitters might then be required to transmit in consecutive slots or else receivers might be required to receive in consecutive slots. (This would depend on whether transmitters of a given station were assigned to consecutive subnetworks and receivers to every Cth subnetwork, as in Fig. 2.6, or vice versa.) In either case, individual transmitters and receivers would have to be potentially active in $\frac{1}{C}$ of the slots.

3.2.2 Hybrid Spatial-Spectral Separation

The main drawback of the spatial separation is the fact that, in an implementation employing a centralized wiring closet, $C_T + C_R$ fibers must be installed between each station and the wiring closet, and a separate coupler must be used for each subnetwork. Similarly, in a linear-bus implementation, C^2 fibers are required, and $2C$ couplers per station. These drawbacks can be obviated if wavelength separation is employed. On the other hand, the number of available wavelengths may also be limited. It is therefore interesting to study the possibility of combining the two. The basic idea is to combine spatial and spectral separation so that any two subnetworks are separated either in space or in wavelength or in both. For brevity, WDM (for wavelength division multiplexing) and SDM (for space division multiplexing) will be used to denote spectral and spatial separation, respectively. The number of different wavelengths will be denoted by W. (A WDM-SDM implementation is obviously also possible with $\mathcal{PBI}$. However, its use with $\mathcal{SBI}$ creates a problem of properly assigning wavelengths to stations, which does not exist with $\mathcal{PBI}$, and is therefore studied in detail.) For clarity of presentation, a wiring-closet implementation will be considered. However, all the results can be adapted to a linear bus implementation by exchanging "couplers" and "fibers". Initially, a scheme which permits a reduction in the number of station-closet fibers will be outlined; the wavelength assignment problem will then be introduced. Lastly, a possible reduction in the number of couplers will be explored.

Reducing the number of fibers

A reduction in the number of fibers connected to each station, while, for the time being, using a separate star coupler for each subnetwork, is achieved as follows. At each transmitting station, outputs of transmitters that use different wavelengths are multiplexed onto a common fiber, thus reducing the number of fibers leading from each transmitting station to the wiring closet. At the wiring closet, demultiplexing takes place, yielding the $N_T \cdot C_T$ distinct signals from all transmitting stations. These are then fed into individual inputs of the couplers, as before. At the output of the couplers, signals of different wavelengths which are destined to the same receiving station are multiplexed onto a common fiber, thereby reducing the number of fibers between the wiring closet and each receiving station. At each receiving station, demultiplexing again takes place, and the single-wavelength signals are connected to individual receivers.

The wavelength assignment problem

Since each of the $C_T \cdot C_R$ couplers represents a subnetwork, it follows that each coupler must carry a single wavelength. Consequently, wavelengths will be assigned to couplers, and the wavelength assigned to a coupler will automatically be assigned to all the transmitters and receivers that are members of the subnetwork represented by that coupler. To minimize the number of fibers, it is necessary (i) to assign $\min(W, C_T)$ and $\min(W, C_R)$ different wavelengths to each transmitting and receiving station, respectively, and (ii) to make equal use of each assigned wavelength at each station; i.e., the numbers of transmitters (receivers) in any given station which use two different wavelengths may differ by at most one. (It is also desirable, for standardization, to assign the same set of wavelengths to all stations.) We next present a simple wavelength assignment algorithm, along with combinations of (W, C_T, C_R) for which it achieves the maximum saving, namely a reduction by a factor of $\min(W, C_T)$ and $\min(W, C_R)$ in the number of fibers connected to transmitting stations and to receiving stations, respectively. A correctness proof is presented in appendix A.

Wavelength-assignment algorithm. arrange the couplers in bunches of G, where G is the least common multiple of C_T and W; number the bunches consecutively modulo W, beginning with 0. Next, number the couplers within each bunch consecutively modulo W, beginning with the bunch's number. The numbers correspond to distinct wavelengths. Applicable combinations:

(1) $W > C_T$, $W \geq C_R$, W divides $C_T \cdot C_R$. (A single fiber is connected to each station. Different stations may be assigned different subsets of wavelengths. Taken to the extreme ($W = C_T \cdot C_R$), this is a pure WDM implementation.)

(2) $W = C_T$ and $C_T > C_R$. (A single fiber is connected to each station. Each transmitting station is assigned all W wavelengths, but different receiving stations may be assigned different subsets of wavelengths.)

(3) W divides C_T. (Each TS is assigned all W wavelengths. Taken to the extreme,

assuming W=C_T=C_R=C, a single fiber will be required from each station to the wiring closet.)

(4) W=C_R, $C_R > C_T$. (A single fiber is connected to each station. Each receiving station is assigned all W wavelengths, but different transmitting stations may have to be assigned different subsets of wavelengths.) This case is the dual of (2), and a slight modification to the algorithm is required: initially, exchange the roles of C_T and C_R and apply the algorithm. Next, let coupler (i,j) be the one that connects TS group i to RS group j in the modified $\mathcal{SBI}$; in the original $\mathcal{SBI}$, use coupler (i,j) to connect TS group j to RS group i. In essence, the problem was converted to case (2), solved and converted back.

(5) W divides C_R. This is the dual of (3). The same procedure as in (4) applies here as well.

(6) Other cases. One way to treat other cases is to augment C_T and/or C_R, also adding the required dummy TS groups and RS groups, so as to reach one of the above cases. This augmentation is such that dummy transmitters and receivers carry higher numbers than the real ones, and dummy station groups have higher numbers than the real ones. As a result, no dummy transmitter or receiver is used to connect a real station with a real station. Also, a coupler that has a dummy group connected to it is a dummy coupler and need not be implemented, so the number of real couplers is not increased. Such a procedure guarantees the optimality in the sense of assigning the maximum possible number of wavelengths to each station. (Including its dummy transmitters or receivers.) As will become apparent in the proof of the optimality of the assignment algorithm, a wavelength is never assigned to transmitters or receivers of any given station a second time before all other wavelengths have been assigned. Therefore, if the dummy transmitters and receivers of any given station are placed after the real ones, (higher numbers,)

a uniform distribution of wavelengths among the *real* transmitters and receivers of any given station will also be achieved.

Reducing the number of couplers

The use of W different wavelengths also permits a reduction by a factor of W in the number of couplers, which is achieved by replacing each group of W couplers that carry different wavelengths with a single coupler. The combination of spatial and wavelength multiplexing can thus be used to support large values of C_T and C_R, while maintaining small cable size and manageable wavelength multiplexing. Some practical aspects of reducing the number of couplers and fibers in a fiber optic implementation will be addressed in a later section.

3.2.3 Other Multiplexing Techniques

The entire previous section is equally applicable to the combination of SDM with a separation in any other domain. Furthermore, if the spatial division is enumerated, (resulting in a (space, wavelength) assignment for each transmitter and receiver,) the same assignment algorithms can be used to support any other combination of two separation domains.

3.2.4 Saving Transmitters and Receivers

Although $\mathcal{SBI}$'s are single-hop, switchless interconnections, the act of selecting which transmitter to use and transferring data from a buffer to the selected transmitter can be viewed as switching. Carrying this one step further, the switching could be done between a single transmitter and the output lines. (For fiber optic implementations, it is worth noting that arrays of 12 individually addressable LED's, directly coupled to multi-fiber ribbon cable, have been developed.[35]) This switching is nevertheless fundamentally different from switching in the interconnection, for several reasons: (i) the switch is located at the TS, so there is no problem of getting power or control to it, (ii) its control is determined solely by the TS at which it resides, (iii) it only affects data that belong to its TS, and (iv) the switching may be very slow without degrading network performance, since it does not utilize any shared resource.

The main limitation in using a single transmitter is that the TS can send at most one message at a time. However, since the actual number of transmitters can be chosen independently for each TS, the implementation of each station can be tailored to its needs and budget. Similar savings in receivers can only be achieved with deterministic access schemes; however, it is possible to replicate the front end, up to and including the address detection, and share the remaining portions. (The throughput-implications of doing this were analyzed in the previous chapter, as the case in which a station could operate at most one of its receivers at any one time.) Finally, it should be noted that the above savings are also possible with $\mathcal{PBI}$, but clearly not with $\mathcal{SBB}$.

3.3 Fiber Optic Implementation

In this section, issues pertaining to the fiber optic implementation of $\mathcal{SBI}$'s are explored, focusing on the single-path, unidirectional, equal-degree $\mathcal{SBI}$ with disjoint subnetworks and grouping. Initially, the passive interconnection component requirements for this $\mathcal{SBI}$ will be compared with those for $\mathcal{PBI}$. Then, the issue of power budget and the number of stations that can be accommodated will be discussed. Lastly, a potentially power-efficient SDM-WDM implementation of the $\mathcal{SBI}$ will be outlined.

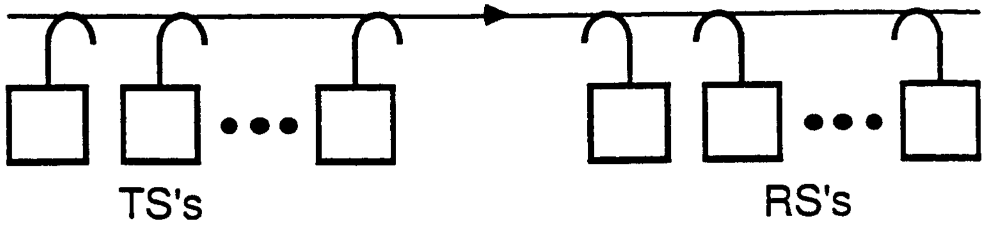


Fig. 3.1 Linear bus implementation of a subnetwork.

3.3.1 Passive Interconnection-Component Requirements

In the previous chapter, we compared the performance of $\mathcal{SBI}$ with that of $\mathcal{PBI}$ in terms of concurrency and delay. The cost was measured only in terms of the number of transmitters and receivers and their speed. In this section, we consider the passive interconnection components, namely couplers and fibers. It will be assumed that those components can operate at any transmission rate. As in the case of performance, the comparison will be conducted for three sets of constraints: (i) equal B and S, (ii) equal C and S, and (iii) equal B and C. Two extreme configurations of an individual subnetwork will be considered: a linear bus with taps, and a centralized star. Throughout the discussion, we assume that there are N stations, each with C transmitters and C receivers.

Linear bus with taps. As shown in Fig. 3.1, each subnetwork is implemented as a single fiber that goes among the stations. Each transmitter is connected to this fiber by means of a (2×2) star coupler, and the same is true of each receiver. For simplicity in comparing the fiber requirements, we assume that each fiber goes among all stations, regardless of whether or not they are members of its subnetwork; however, stations that are not members of a given subnetwork are obviously not connected to its fiber.

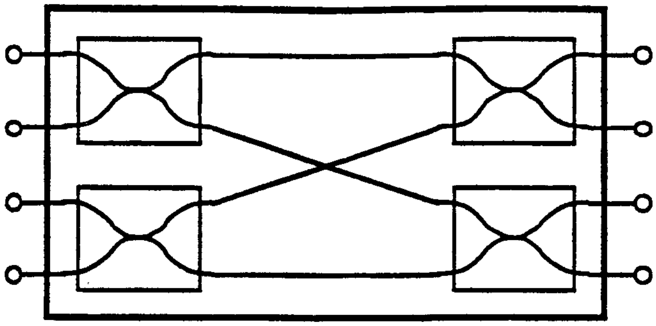


Fig. 3.2 A (4 × 4) star coupler constructed from four (2 × 2) couplers.

Centralized star. As shown in Fig. 2.6, this is the dual of the linear bus. Here, a star coupler corresponds to a subnetwork, and a fiber corresponds to a transmitter or a receiver.

The comparison of the interconnection component requirements is complicated by the fact that the required star couplers are of different sizes. We solve this by assuming that large couplers are implemented using small ones as building blocks [36]. This is illustrated in Fig. 3.2, which depicts a (4×4) coupler constructed using 4 couplers of size (2×2). In general, an $(M \times M)$ coupler can be constructed using $\frac{M}{p} \cdot \log_p M$ couplers of size $(p \times p)$. (It is assumed that p divides M.)

Table 3.1 summarizes the comparison. Perhaps the most interesting result is that for equal B and C, (the case in which $\mathcal{SBI}$ has higher throughput for identical active hardware,) and a star configuration, $\mathcal{SBI}$ requires fewer couplers and the same amount of fiber.

Linear bus implementation:

$$\text{Fiber} \longleftrightarrow \text{Subnetwork} \longleftrightarrow S/B$$

$$\text{Star Coupler} \longleftrightarrow \text{Transmitter or Receiver} \longleftrightarrow C$$

Conditions:	Fiber		Couplers	
equal	$\mathcal{PBI}$	$\mathcal{SBI}$	$\mathcal{PBI}$	$\mathcal{SBI}$
S and B	S/B	S/B	$2N \cdot C_{\mathcal{PBI}}$	$2N \cdot \sqrt{C_{\mathcal{PBI}}}$
C and B	C	C^2	$2NC$	$2NC$
S and C	C	C^2	$2NC$	$2NC$

Centralized star implementation:

$$\text{Fiber} \longleftrightarrow \text{Transmitter or Receiver} \longleftrightarrow C$$

$$\text{Star Coupler} \longleftrightarrow \text{Subnetwork} \longleftrightarrow S/B$$

Conditions:	Fiber		Couplers	
equal	$\mathcal{PBI}$	$\mathcal{SBI}$	$\mathcal{PBI}$	$\mathcal{SBI}$
S and B	$2NC_{\mathcal{PBI}}$	$2N\sqrt{C_{\mathcal{PBI}}}$	$C_{\mathcal{PBI}}\ (N \times N)$	$C_{\mathcal{PBI}}\ \left(\frac{N}{C_{\mathcal{PBI}}} \times \frac{N}{C_{\mathcal{PBI}}}\right)$
C and B	$2NC$	$2NC$	$C \quad (N \times N) =$	$C^2 \quad \left(\frac{N}{C} \times \frac{N}{C}\right) =$
			$\frac{NC}{p}\log_p N\ (p \times p)$	$\frac{NC}{p}\log_p \frac{N}{C} \quad (p \times p)$
S and C	$2NC$	$2NC$	"	"

Table 3.1. Passive interconnection-component requirements for $\mathcal{SBI}$ and for $\mathcal{PBI}$. (Shown for both a linear bus and a centralized star implementation.)

3.3.2 Path Loss and the Number of Stations that Can Be Accommodated

3.3.2.1 Path Loss

Path loss is the ratio of the power at the output of a transmitter, P_T, and the power at the input of a receiver, P_R. Its constituents are:

- **Power split.** Since coherent detection is not practical in FOLAN's, and since the optical detectors in use have low impedance, the receivability of a signal depends on its power level, and the reception of a signal "consumes" the power that is present at the receiver's input. This is in contrast with the case of coaxial cables and high impedance detectors, which sense the voltage and draw minimal amounts of power. As for the star couplers used in the interconnection, the power that appears at any given input of a coupler is split among the outputs, and the power at each output is only a fraction of the power at the input.
- **Inefficient fan-in.** If fibers of constant cross section are used, an $(m \times n)$ lossless coupler has a power split of $\max\{m, n\}$.[†] (The ratio of power at a single output to that at the input is $\max\{m, n\}$.)
- **Excess loss.** This represents the imperfection of the coupler and its connectors.

For a given value of P_T, the maximum allowable path loss is determined by $P_{R_{\min}}$. In studying the performance of existing optical receivers, one observes that, over a wide range of transmission rates (100MHz-1GHz), $P_{R_{\min}}$ is roughly proportional

[†] This is indirectly explained by the constant radiance theorem in optics,[37] which states that when a narrow beam undergoes a linear lossless transformation, its radiance remains constant. A corollary of this is that the product of the cross-sectional area and the square of the numerical aperture of an optical beam must remain constant under any lossless linear transformation of that beam.[38] As a result, when several fibers are fused to form a single fiber, as is the case at the input of a star coupler, the cross-sectional area decreases and the numerical aperture increases. Unfortunately, the numerical aperture of the fiber is not any larger than that of the original one, so most of the power cannot propagate and is lost. The fact that the cross-sectional area again increases at the output of the coupler does not help.

to the transmission rate. In other words, the receiver requires a certain number of photons (a fixed amount of energy) per bit [39]. As a result,

$$P_{R_{\min}}(B) = B \cdot P_{R_{\min}}(1). \tag{3.1}$$

3.3.2.2 Maximum Number of Stations

The number of stations that can be accommodated by a passive fiber-optic interconnection is determined by the maximum path loss over all source-destination pairs. Since the subnetworks are disjoint, the first step in determining the maximum number of stations is to derive the maximum number *per subnetwork* as a function of the permissible path loss. The total number of stations is a simple function of this number. Two configurations will be considered: a linear bus with taps, and a centralized star.

Linear bus with taps

In the linear bus configuration, a signal goes through a number of couplers which is proportional to the number of stations in the subnetwork, N_{sn}. For simplicity, all couplers are assumed to be identical, so that the ratio of the power on the bus just before any given coupler and that immediately after it is constant. Let L, ($L \geq 1$,) denote the contribution of excess loss and fan-in loss to this ratio; the contribution of the power split to the path loss will be lumped together and is equal to N_{sn}. This isolates the configuration-independent component, (fan-out,) from the configuration-dependent ones. Finally, it should be noted that there is a tap for each transmitter and for each receiver on the subnetwork; nevertheless, since we are interested primarily in the comparison between $\mathcal{SBI}$ and $\mathcal{PBI}$, we will assume that a signal goes through exactly N_{sn} taps.[†] Letting P_T and P_R denote the power at the output of a transmitter and the power at the input of a receiver, respectively, we obtain:

$$\frac{P_T}{P_R} = N_{sn} \cdot L^{N_{sn}}, \tag{3.2}$$

and the number of stations that can be accommodated is such that

$$N_{sn} + \log_L N_{sn} = \log_L \left(\frac{P_T}{P_{R_{\min}}} \right). \tag{3.3}$$

The first term on the left hand side dominates the second one when the loss dominates the power split, and the second term dominates the first one when the power split dominates the loss.

Star configuration

The star configuration is logically an $(N_{sn} \times N_{sn})$ star. However, since very large stars are not available, we assume that the large star is implemented using elementary $(p \times p)$ stars as building blocks. Consequently, the signal goes through $\log_p N_{sn}$ couplers on its way from the transmitter to any receiver. The path loss is hence

$$\frac{P_T}{P_R} = N_{sn} \cdot L^{\log_p N_{sn}} = N_{sn}^{(1+\log_p L)}, \tag{3.4}$$

and the number of stations that can be accommodated is

$$N_{sn} = \left(\frac{P_T}{P_{R_{\min}}} \right)^{\frac{1}{1+\log_p L}}. \tag{3.5}$$

[†] With reciprocal couplers, the fraction of power that is coupled from a transmitter onto the bus is equal to the fraction that is taken off the bus to the dangling output of the coupler. This creates a tradeoff in the selection of the coupling coefficient, which does not exist in the couplers that act as taps for receivers. Consequently, the coupling coefficients in the receiver couplers are likely to be different from those of the transmitter couplers. We therefore think of each pair of transmitter and receiver couplers as a single coupler, whose loss accounts for the losses in its two constituents.

Comparison

The comparison will be under the constraint of equal aggregate network throughput and equal numbers of transmitters and receivers per station. (Equal S for all configurations; also, equal C for $\mathcal{SBI}$ and $\mathcal{PBI}$.) As a result, $\mathcal{SBI}$ will require a lower transmission rate.

In a linear bus configuration implemented using currently available components, the loss (fan-in and excess loss) dominates the power split. Consequently,

$$N_{sn} \approx \log_L \left(\frac{P_T}{P_{R_{\min}}(1) \cdot B} \right). \tag{3.6}$$

In the star configuration, the opposite is true, and therefore

$$N_{sn} \approx \frac{P_T}{P_{R_{\min}}(1) \cdot B}; \tag{3.7}$$

Let N_0 denote the maximum number of stations that can be accommodated by a single broadcast bus that achieves the desired throughput. Since the transmission rate required by $\mathcal{PBI}$ in order to achieve a given throughput is C times lower than that required by a single bus, and the rate required by $\mathcal{SBI}$ is C times lower than that of $\mathcal{PBI}$, the relationship between the number of stations that can be accommodated on each subnetwork with each of the schemes is as follows.

Linear bus with taps:

$$N_{sn}^{\mathcal{SBI}} \approx N_0 + 2\log_L C$$

$$N_{sn}^{\mathcal{PBI}} \approx N_0 + \log_L C \tag{3.8}$$

Star configuration:

$$N_{sn}^{\mathcal{SBI}} \approx C^2 \cdot N_0$$

$$N_{sn}^{\mathcal{PBI}} \approx C \cdot N_0 \tag{3.9}$$

Finally, recall that with a single broadcast bus as well as with $\mathcal{PBI}$, $N = N_{sn}$, whereas with $\mathcal{SBI}$, $N = C \cdot N_{sn}$. The relationship between the total number of stations that can be accommodated by the different interconnections is therefore as follows.

Linear bus with taps:

$$N^{\mathcal{SBI}} = C \cdot (N_0 + 2\log_L C);$$

$$N^{\mathcal{PBI}} = N_0 + \log_L C. \tag{3.10}$$

Star configuration:

$$N^{\mathcal{SBI}} = C^3 \cdot N_0;$$

$$N^{\mathcal{PBI}} = C \cdot N_0. \tag{3.11}$$

The maximum number of stations which can be accommodated by $\mathcal{SBI}$ is thus always higher than the corresponding numbers for the single bus or $\mathcal{PBI}$ by at least a factor of C, due to the fact that $N^{\mathcal{SBI}} = C \cdot N_{sn}^{\mathcal{SBI}}$. An additional advantage of up to C^2 over the single bus and up to C over $\mathcal{PBI}$ is a byproduct of the reduced transmission rate. This factor, however, depends on the configuration of the subnetworks. Unfortunately, it is least pronounced for a linear bus, which is the configuration that can accommodate the least number of stations. It should be noted that, unlike the performance advantage, the power advantage is independent of traffic pattern. Also, the advantage increases as the transmission rate increases. It is thus of particular interest for high speed networks.

Finally, note that this $\mathcal{SBI}$, when implemented as just described, is optimal in terms of power budget. This is so because the total fan-out of each transmitter is the bare minimum, (if C_T transmitters must reach N stations, at least one transmitter must reach $\frac{N_R}{C_T}$ receivers,) fan-in does not add to the loss, provided that $\frac{N_T}{C_R} \leq \frac{N_R}{C_T}$, and the number of stages is minimal, thus minimizing the excess loss. Unfortunately, however, this cannot be said of any $\mathcal{SBI}$. A general equal-degree, single-path $\mathcal{SBI}$ may require each signal to go through a $(1 \times \frac{N_R}{C_T})$ coupler, followed by a $(\frac{N_T}{C_R} \times 1)$ coupler, with a total fan-in and fan-out loss of $\frac{N_T \cdot N_R}{C_T \cdot C_R}$.

3.3.3 Efficient Fiber Optic Implementation of an $\mathcal{SBI}$ that Achieves a concurrency of $1.5 \cdot C^2$ for a Uniform Traffic Pattern

The $\mathcal{SBI}$ that achieves $1.5 \cdot C^2$ has grouping only in the weak sense, and its subnetworks are not disjoint. Nevertheless, it will now be shown how to construct it without any fan-in loss and with optimal fan-out. This $\mathcal{SBI}$ can be viewed as consisting of $9 \cdot C^2$ partially overlapping subnetworks, each interconnecting one group of TS's with one group of RS's. (Recall that there are $3 \cdot C$ groups of each type.) Fig. 3.3 illustrates a power-efficient configuration for this $\mathcal{SBI}$, which is constructed as follows. The outputs of the ith transmitters of all stations in any given TS group are fed into an $(\frac{N_T}{3C} \times \frac{N_T}{3C})$ star coupler. Next, noting that the aforementioned transmitters must reach 3 groups of RS's, the outputs of the star coupler are split into 3 bundles of $\frac{N_T}{9C}$ fibers each.(All the fibers in a bundle carry identical information.) Now, let us examine the situation at a group of RS's; for the jth receivers, there are 3 incoming bundles, each with $\frac{N_T}{9C}$ fibers, giving a total of $\frac{N_T}{3C}$ fibers.

Since the signals on those fibers must reach $\frac{N_R}{3C}$ receivers, an $(\frac{N_T}{3C} \times \frac{N_R}{3C})$ star coupler is used. However, since the $\frac{N_T}{3C}$ incoming fibers constitute $\frac{N_T}{9C}$ replicas of

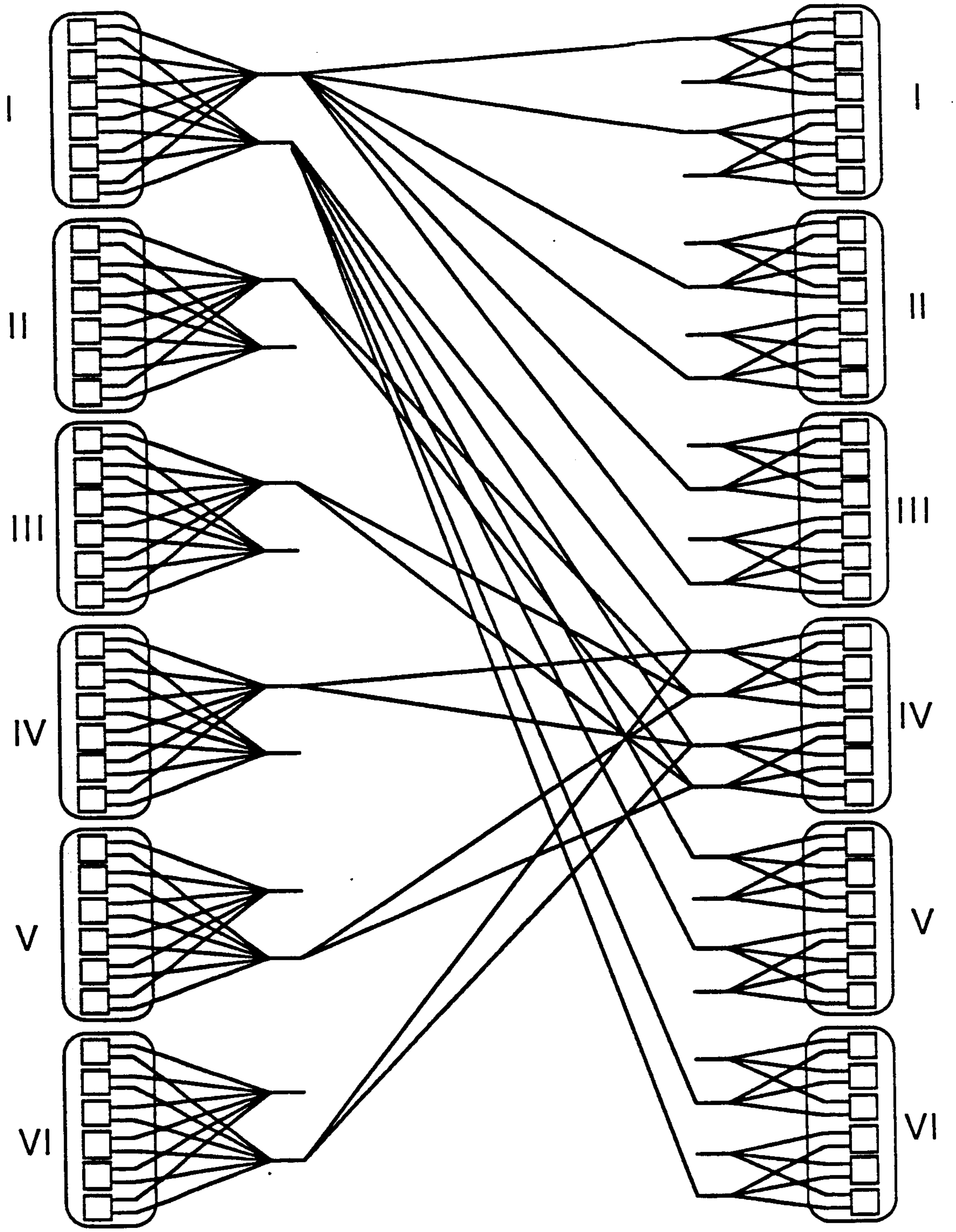


Fig. 3.3 Power-efficient fiber optic configuration for the $\mathcal{SBI}$ that achieves $1.5 \cdot C^2$ with a uniform traffic pattern. The connections are shown for the 1st TS group and for the 4th RS group.

each of 3 signals, it is possible to use $\frac{N_T}{9C}$ couplers of size $(3 \times 3\frac{N_R}{N_T})$ in parallel instead of the single $(\frac{N_T}{3C} \times \frac{N_R}{3C})$ star coupler. This does not increase the number of stages, and prevents interference problems which may occur whenever several replicas of the same signal follow different paths and are then merged. (Multipath.) In summary, this is a two-stage interconnection; i.e., a signal must go through two star couplers on its way from source to destination. The first stage consists of $3C^2$ couplers of size $(\frac{N_T}{3C} \times \frac{N_T}{3C})$, and the second stage consists of $\frac{N_T C}{3}$ couplers of size $(3 \times \frac{3N_R}{N_T})$. (Each stage can be implemented using several stages of smaller couplers, but this is not a requirement.) Recalling the requirements for power optimality, we observe that it is achieved only if $N_T \leq N_R$, as was the case with grouping. It should be noted that, strictly speaking, the power budget is somewhat inferior to that in the case of grouping and disjoint subnetworks, since each path traverses at least 2 couplers, thus incurring an excess loss at least twice. However, this is not a very big difference and, whenever large couplers are constructed from small ones, power budget will be the same.

3.3.4 SDM-WDM Implementation

In fiber optic implementations using different wavelengths as the means of separating subnetworks, it is necessary to multiplex signals of different wavelengths onto a common fiber and, at a later stage, to separate them. One way to do this is to use couplers. However, this is costly in terms of power. Furthermore, the efficient merger of signal of different wavelengths, each of which is on a separate fiber, onto a common fiber, is theoretically possible, contrary to the case of a common wavelength.

At a transmitting station, integrated laser arrays could be used. In fact, five different wavelengths have reportedly been multiplexed onto a single fiber from an integrated laser array,[40], and ten from separate lasers [41].

In a centralized star implementation, a diffraction grating could be used to perform the demultiplexing and multiplexing. Fig. 3.4 depicts a possible implementation for the case $W=C_T=C_R=C$, wherein N multi-wavelength fibers, each of which carries the C outputs of some TS, serve as the input to a single grating spectrometer, which demultiplexes them and outputs $C \cdot N$ signals. Each such signal enters a fiber, which serves as an input to one of the C^2 couplers. The coupler outputs are fed back to the spectrometer, (which is now used in the reverse direction,) such that all signals that should reach a given RS form a horizontal row. The spectrometer multiplexes each row onto a single fiber. The figure depicts the incoming fibers, the grating, two representative $(\frac{N}{C} \times \frac{N}{C})$ couplers, the coupler outputs being fed back to the spectrometer, and the fibers going from the spectrometer to the RS's. Since C different wavelengths are being used, the number of couplers can also be reduced by using an additional grating. However, this may not be economical due to the additional gratings and alignment stages that are required.

It is very important to note that the wavelength–coupler tradeoff affects the number of couplers but has no effect on the size of each coupler, and that efficient merging of different wavelengths is possible. Consequently, the power budget is not affected. Also, the effect on hardware requirements is the same for $\mathcal{PBI}$ and for $\mathcal{SBI}$; consequently, the previous comparison is valid regardless of implementation.

3.4 Operation of an $\mathcal{SBI}$

The discussion in this section will address the operation of $\mathcal{SBI}$'s with nonideal access schemes from several angles. Initially, access schemes will be classified in a rather crude way, and their effect on the actual performance difference between $\mathcal{SBI}$ and $\mathcal{SBB}$ will be discussed. For the purpose of this comparison, $\mathcal{SBI}$ will be treated

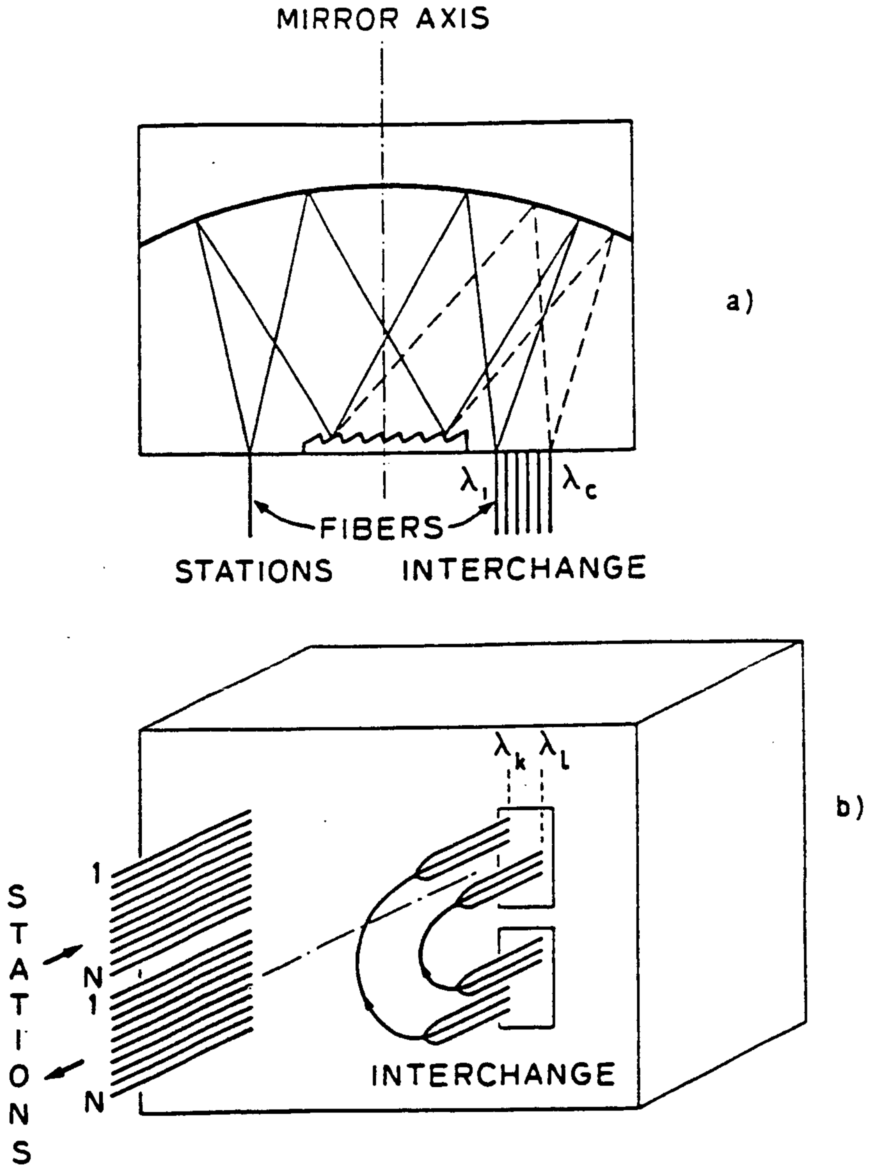


Fig. 3.4 Using a grating spectrometer in a power-efficient SDM-WDM implementation of a unidirectional, single-path, equal-degree $\mathcal{SBI}$ with grouping and disjoint subnetworks. The figure illustrates the case wherein only a single fiber is required between each TS or RS and the wiring closet, but the number of couplers is not reduced. The grating spectrometer is used only to demultiplex signals that arrive from TS's, and to multiplex them onto fibers that go to RS's.

as a collection of C^2 identical, independent subnetworks. Next, it will be shown that to actually operate $\mathcal{SBI}$ using certain access schemes, some modifications must be made to the interconnection. Lastly, ways of jointly operating the entire $\mathcal{SBI}$ will be outlined.

3.4.1 Access Schemes for Sharing a Single Channel

Over the last two decades, much research has been devoted to finding schemes for the efficient sharing of the broadcast bus. Initially, the ALOHA scheme was proposed and implemented [42]. The idea is that a station transmits whenever it has a packet for transmission. If it does not receive an acknowledgment within a certain time, it reschedules a transmission to some random time in the future. The scheme is very simple, but the maximum throughput is only $B/2e$. This was the first of a class of schemes which are often referred to as "random access" schemes, since the allocation of channel time to stations follows no particular order. The next step was to add channel-sensing capability, so that a station does not begin transmission on an already busy channel. (This would lead to mutual destruction of all colliding transmissions.) The first scheme in this category was "Carrier-Sense Multiple Access", or CSMA [43]. The next step was the addition of a collision-detection capability, for the purpose of minimizing the waste in the event of a collision. Such a scheme, known as CSMA-CD, became very popular and is used in the most prominent example of a network employing $\mathcal{SBB}$, Ethernet [8].

The nonzero propagation delay on the channel may cause several stations to believe the channel to be idle and to start transmitting, thereby causing a mutually-destructive collision. The severity of this problem increases with an increase in the ratio of end-to-end propagation delay to packet transmission time, which is commonly denoted by a. For small values of a, the channel-sensing access schemes achieve very high utilization. However, as a becomes large, increasing the transmission rate results only in a marginal increase of channel capacity. In fact, for very high speed networks, the performance of channel-sensing random-access schemes can be worse than that of random transmissions without channel sensing, such as ALOHA. In [44, 45], it is shown that the maximum channel throughput for the infinite population, slotted CSMA-CD scheme is given by

$$S = \begin{cases} \dfrac{B}{1+Ka} & a \leq 0.5, \\ \dfrac{B}{(2+K)a} & a > 0.5, \end{cases} \tag{3.12}$$

where K is a constant in the neighborhood of 3 to 6 which depends on the particular version of the protocol. (Recall that $a \propto B$.)

A drawback which is common to all random access schemes is that, due to their stochastic nature, they cannot guarantee delivery of a message within a specified time, even when the offered traffic is less than channel capacity.

In recent years, demand-assignment multiple access (DAMA) schemes have emerged [46]. With these schemes, the right to transmit is passed from user to user in a round robin manner, thereby guaranteeing bounded delay. However, unlike in the original time-division multiple-access schemes, a user that has no messages to transmit wastes only a minimal amount of channel time. Some of the schemes, such as Expressnet [45] and Fasnet [47], use the physical ordering of the stations along the bus to determine the order of transmissions. This permits them to operate more efficiently than random-access schemes at high speeds, since they incur a significant time gap in transmissions (twice the end-to-end propagation delay on the bus) only once per round. (Eventually, however, they too become inefficient.) Another nice feature of these schemes is that as the number of active users increases, there are more transmissions per round, and the channel utilization thus increases.

3.4.2 Effect of the Access Scheme on the Relative Performance of $\mathcal{SBI}$ and $\mathcal{SBB}$

Consider a slightly futuristic example of a local-area network with a total bandwidth of 1Gbps, bus length of 1.5km, (end-to-end propagation time of approximately 10μsec, and packet length of 2500 bits. With $\mathcal{SBB}$, $a = 4$, and the maximum throughput with slotted CSMA-CD is less than 50Mbps; i.e., a utilization of 0.05. In contrast, an $\mathcal{SBI}$ with $C = 3$ would have a transmission rate of only 110Mbps, resulting in $a = 4/9$ and $S >$ 400Mbps; i.e., an improvement in channel utilization by more than a factor of 8, in addition to the other advantages. Furthermore, with such an advantage in throughput, (and note that this the case that was referred to in the theoretical chapter as equal S and C,) it is clear that for all cases, except that of extremely low load, the delay performance of $\mathcal{SBI}$ will also be far superior. The conclusion is that $\mathcal{SBI}$ can extend the effective range of random access schemes and permit their efficient use in situations that would otherwise render them useless.

At the other extreme, the performance of ALOHA is invariant under a, so the relative performance depends only on the topology and the results of the previous chapter apply. DAMA schemes are between the two extremes, and a detailed comparison is omitted. However, it is clear that increasing a, i.e. going to high speed, longer range networks, favors $\mathcal{SBI}$, and the results of the previous chapter serve as a lower bound on the advantage of $\mathcal{SBI}$.[†]

3.4.3 Operation of an $\mathcal{SBI}$ Subnetwork with Various Access Schemes

The operation of an $\mathcal{SBI}$ subnetwork is straightforward if the $\mathcal{SBI}$ is bidirectional. It is also straightforward if the access scheme being used does not require

[†] The performance of DAMA schemes often increases as the number of active users increases, since the round time increases and the overhead remains fixed. The fact that in $\mathcal{SBB}$ all stations participate in the same round, whereas in $\mathcal{SBI}$ only a subset transmits over each bus, favors $\mathcal{SBB}$. However, this can never offset the fact that a is lower for $\mathcal{SBI}$ by a factor of C^2.

the sensing of the channel. ALOHA and CDMA are such schemes. (CDMA was discussed in chapter 1.) However, operating a subnetwork of a unidirectional $\mathcal{SBI}$ with a channel-sensing access scheme is complicated by the fact that a station (we think of each TS, RS pair as a single bidirectional station) can only hear one of the C_T channels over which it may transmit. With bidirectional media, the problem can be solved by equipping each transmitter with a sensor, which is usually much cheaper than a receiver. With unidirectional media, the signal has to be brought to the transmitter's location. In a centralized star implementation, this would require extra wires, resulting in a 50% increase in the amount of wiring. (Two wires per transmitter, one per receiver.) A cheaper option is available if, as is the case with fiber optics, the medium is unidirectional only in the sense that propagation in the two directions is independent, but signals in both directions can be handled concurrently. In such a case, it is possible to reflect a fraction of the power from the output of the star coupler back to the transmitters. For example, if the coupler has an extra output, a mirror could be placed at the end of that fiber.

3.4.4 Combined Operation of the Entire $\mathcal{SBI}$

The combined operation of all subnetworks of an $\mathcal{SBI}$, in particular multiple-path $\mathcal{SBI}$'s, opens an avenue for much future research. The discussion here, however, is restricted to several preliminary observations.

- With a unidirectional, single-path $\mathcal{SBI}$ with disjoint subnetworks and grouping, the intragroup subnetworks, say (i,i), can be used by the members of the group for the coordination of their transmissions over all subnetworks of which they are transmitting members, (i,j).
- Adding a single control bus constitutes a relatively small overhead. $(1/C^2.)$
- In a hybrid $\mathcal{SBI}$-$\mathcal{PBI}$, a reasonable policy will be to first try and use the $\mathcal{SBI}$,

resorting to the $\mathcal{PBI}$ only in case of failure or congestion of the relevant subnetwork.

- In the summary of chapter 2, it was noted that a unidirectional $\mathcal{SBI}$ can be used in both directions, providing bidirectional connectivity between two groups of bidirectional stations, but no connectivity among the members of each group. To operate an $\mathcal{SBI}$ in this mode, one can make use of the corollary to property 2 of interference. The corollary states that two paths are mutually noninterfering if an only if the "reverse" paths are mutually noninterfering. Consider now a slotted access scheme, in which odd-numbered slots are used for the generation of requests, and even-numbered slots are used to reply or to acknowledge requests that were generated in the previous slot. The implication of the corollary is that, regardless of the scheme used to regulate the use of the odd-numbered slots, replies will always succeed. (Note that a reply is always destined to the station that generated the request, and that replies are only sent for successfully received requests. Therefore, the set of reply paths consists of the reverse paths corresponding to a set of mutually noninterfering paths.)
- The possibility of implementing a TS using a single physical transmitter was mentioned in an earlier section. With random-access (e.g. CSMA-CD) or *implicit* token-passing (e.g. Expressnet) schemes, the operation of such a TS would be quite simple: upon selection of the subnetwork over which it wishes to transmit, it would attach its transmitter (and sensor) to that subnetwork and begin executing the access protocol. However, operation in conjunction with schemes in which high overhead is associated with the addition or removal of a node (e.g. Token Bus, TDMA) would be complicated and inefficient.

High level protocols, such as maintaining address tables and topological information, are beyond the scope of this research.

3.5 Applicability

Having explored the theory of $\mathcal{SBI}$ in chapter 2, and practical issues pertaining to its implementation and operation in this chapter, it is now possible to discuss the applicability of $\mathcal{SBI}$ to various environments. The discussion will focus on the single-path, unidirectional $\mathcal{SBI}$ with disjoint subnetworks and grouping, and all implied comparisons will be with $\mathcal{SBB}$. ($\mathcal{PBI}$ is left out, since the purpose of this discussion is to compare $\mathcal{SBI}$ with the most commonly used single-hop interconnection, which is clearly $\mathcal{SBB}$.)

3.5.1 Bus-Oriented Local-Area Networks

Under a constraint of fixed total communication bandwidth, $\mathcal{SBI}$ offers increased throughput, (due to the increased efficiency of the access scheme as a result of a C^2-fold reduction in a,) increases the utilization of transmitters and receivers, is optimal in terms of power budget, and permits a transmission rate that is lower by a factor of C^2 than the total transmission bandwidth. However, packet transmission time is C^2 times longer, throughput depends on the traffic pattern, and C transmitters and receivers are required for each station.

Under a constraint of fixed transmission bandwidth per station, $\mathcal{SBI}$ again offers higher throughput: an increase by a factor of C due to increased total communication bandwidth, as well as an increase in efficiency due to a C-fold reduction in a. Transmission time is higher only by a factor of C.

Under a constraint of equal transmission rates (B), $\mathcal{SBI}$ offers C^2-fold higher throughput while requiring only C times more transmitters and receivers. Efficiency is equal, and delay is lower for $\mathcal{SBI}$.

The appropriateness of each of the above constraints depends on the implementation, as is suggested by the following observations, which also expose additional advantages of $\mathcal{SBI}$ and play down the practical importance of some of its apparent disadvantages.

- In existing networks, users often cannot make use of the total bandwidth. For example, transferring a long file from one Vax 11/750 (running Unix 4.2BSD) to another over an unloaded 10Mbps Ethernet proceeds at an average rate of approximately 3.3Mbps. In this case, the user would not feel a difference in the transfer rate if it used one of three transmitters, each of which is capable of transmitting 3.3Mbps. In other words, the bottleneck is the user's capability to move data into the transmission buffer or out of the reception buffer, rather than the transmission rate.
- The cost of transmitters and receivers is constant within certain ranges of transmission rates. However, the general trend is a faster than linear increase. (The cost becomes infinite as the feasibility boundary is reached.) Therefore, several slow transmitters and receivers may actually cost less than a single pair whose transmission rate is equal to the sum of the slow ones.
- The total communication bandwidth of a medium is sometimes a function of the way in which it is used. Consider, for example, a coaxial cable. Its bandwidth for a single base-band channel is on the order of 10Mbps. Yet, CATV cables carry over 100 channels, each of 5MHz bandwidth. The difference is due to dispersion; (i.e., a variation of propagation speed with frequency;) this causes a single wide-band signal to become badly distorted,[†] whereas each of numerous narrowband signals suffer no noticeable distortion. It follows that, with a single cable and frequency-separation, $\mathcal{SBI}$ can have a total communication bandwidth on the

[†] On a point-to-point link of known length, dispersion can be compensated for using a frequency-dependent delay line. In the case of networks, however, particularly with a linear bus implementation, the distance depends on the source-destination pair. This would require either adaptive compensation at the receiver, or the use of compensators along the bus. (In fact, dispersion-compensated twisted pair wire has been developed.)

order of C^2 times higher than that of $\mathcal{SBB}$.

The implication of all the above is that $\mathcal{SBI}$ is quite promising for situations in which a single, inexpensive bus cannot satisfy the requirements; the exceptions are cases wherein the primary performance measure is transmission time or delay at very low loads, or a traffic pattern which has severe, time varying nonuniformities. Finally, it should be noted that $\mathcal{SBI}$ can also serve as a backbone network, in which at least some of the stations serve as concentrators for small users. (As do tips for terminals on an Ethernet.)

3.5.2 Other Networks

$\mathcal{SBI}$ may also be useful for radio networks. Consider, for example, a task force which is equipped with simple voice radios. With a single radio for each member, at most one person can talk at any given time. If, instead, each person is equipped with 2 radios, or two receivers and one frequency-agile transmitter, the task force can use 4 channels, thereby permitting 4 ongoing conversations. The important difference between this example and the LAN is that here it is not simple to have a single fast channel. (Sophisticated radios would be required.) Although the idea of a single person having two radio transceivers sounds somewhat unusual, observation shows that policemen quite often do just that. The difference between this and a conventional multi-channel network, as is found in the military, is that here there is full connectivity, whereas military networks provide communication between a node and its subordinates on one channel and between a node, its siblings and its superior on a second channel. (Tree structure.)

In satellite networks, it is often desirable to divide the bandwidth into several channels, each of which is operated using TDMA. Viewing those channels as subnetworks of an $\mathcal{SBI}$ would require each station to tune to only $1/C$ of the channels, yet would guarantee connectivity.

3.5.3 Memory-Processor Interconnection

Performance

$\mathcal{SBI}$ cannot compete with multi-hop, high-flux interconnections in terms of cost-performance ratio. It is therefore not a good solution when massive parallelism is required in the communication. At the same time, however, many small-scale parallel machines tend to continue using a single bus. $\mathcal{SBI}$ may be appealing for a range of machines in which a single bus is insufficient, yet a high-flux interconnection is an expensive and complicated overkill. $\mathcal{SBI}$ will provide a significant throughput enhancement, while retaining the simplicity of a bus. For example, if the 32-bit ports of devices that are currently connected to a single 32-bit bus are replaced with two 16-bit ports, four 16-bit buses can be constructed, thereby doubling the total communication bandwidth. As in the case of networks, however, $\mathcal{SBI}$ is not good if the main performance measure is latency, queueing delay and throughput not being a problem.

Hardware utilization

Another important aspect in this case is the hardware utilization. The performance of VLSI devices and the degree of integration that can be achieved are often determined by the limited number of pins that a device may have. If the utilization of each pin is doubled, the number of pins can be cut in half. (Not exactly, since the control pins, which must be replicated for each port, were not counted.)

Chapter 4

"Supernodes" in Networks Employing Spread-Spectrum with Code-Division Multiple-Access

4.1 Introduction

The two previous chapters focused on the problem of increasing the capacity of single-hop interconnections that employ shared media. The increase was accomplished by the use of multiple channels, each of which interconnected a subset of stations. To preserve the single-hop connectivity, it was necessary to equip each station with several transmitters and receivers. While the discussion implicitly assumed narrowband channels, the results are equally applicable to spread-spectrum channels. The current chapter considers a single spread-spectrum channel, which interconnects a set of nodes.[†] The interconnection is thus an $\mathcal{SBB}$, although this

[†] In the $\mathcal{SBI}$ chapters, the term "stations" was used in order to avoid confusion with the graph nodes, since those corresponded to single transmitters and receivers or to couplers. Here, there is no confusion and the more common term, "nodes", is thus used.

channel could also be one of an $\mathcal{SBI}$'s subnetworks. Equipping a station with several transmitters and receivers that are connected to the same channel cannot increase the interconnection's capacity; nevertheless, it is a means of achieving nonuniform capacity allocations so as to match nonuniform traffic patterns. In the remainder of this section, we present an overview of spread spectrum, explain the capacity allocation problem and outline the contents of the remainder of the chapter.

4.1.1 The Spread-Spectrum Channel with CDMA

At the transmitter, a data stream of rate B bits per second is used to modulate a data-independent *code*-stream of a much higher rate, $SF \cdot B$. The resultant *chip*-stream, of rate $SF \cdot B$, is used to modulate the carrier of the transmitter. The bandwidth occupied by the transmission is thus approximately SF times higher than the minimum bandwidth required for transmitting data at rate B. SF is therefore referred to as the *spread factor*.

Two basic mechanisms are in use for achieving the spreading, and existing systems use either one or a combination. With *frequency hopping, (FH,)* the code determines the carrier frequency for each chip, and the data determines the phase. With *direct sequence, (DS,)* both the data stream and the code are binary sequences. An exclusive-OR operation is performed between the two, and the result is used to modulate the phase of a fixed-frequency carrier.

At the receiver, the chip-stream is received. Then, using the code, which is also known to the receiver, the original data stream is recovered. (A correlator or a matched filter are employed to perform this operation.) In an ideal situation, the received and transmitted chip streams are identical, and the recovered data stream is therefore identical to the original one. In a non-ideal case, the received signal is contaminated with noise, as well as with other ongoing transmissions. This results in occasional errors in the received chip stream. However, due to the large number of chips per bit, the bit can be reconstructed correctly even if some of the chips are erroneous. Furthermore, it is possible to add error-correction encoding prior to the spreading, and decoding after the despreading. Referring to the elements in the post-encoding (and pre-decoding) streams as symbols, a certain rate of errors in the received symbols can also be tolerated.

In addition to improving immunity to noise and jamming, the use of spread spectrum also permits several concurrent transmissions to be received by collocated receivers. For this to be true, the code must have narrow autocorrelation mainlobes and low sidelobes; this causes replicas of the same code which are staggered in time to be nearly orthogonal to each other, resulting in one transmission having little effect on the output of a correlator which is locked onto another one. The ability to receive several transmissions on the same code concurrently is referred to as *time capture*. Another possibility is to use different, mutually orthogonal codes for each transmission. This is referred to as *code-division multiple-access*, or CDMA. A very large amount of research has and is being done to further the understanding of the performance of various spread-spectrum channels. Some examples are [24, 48, 49, 50].

4.1.2 The Problem in Allocating Capacity

In a real network, certain nodes must often carry much more traffic than most other nodes; examples of such nodes are gateways, mainframes and file servers in terrestrial networks, as well as the terrestrial hub of a 2-hop satellite network [51]. To make use of a channel's capacity, it is necessary to allocate fractions of it to nodes according to the fraction of traffic that they must carry, as specified by the traffic pattern. A "busy" node must therefore receive a fraction of channel capacity which is much larger than those given to most other nodes.

From the fact that the spread-spectrum channel can accommodate several ongoing transmissions, it immediately follows that a single transmission uses only a fraction of the channel capacity. Assuming the use of standard equipment and codes of equal rates for all transmissions, extreme nonuniformity in capacity allocation can therefore only be achieved by equipping a "busy" node with several transmitters and receivers, thereby permitting it to engage in concurrent transmissions or receptions. Such a node will be referred to as a "supernode".

4.1.3 Outline of the Chapter

Having established the need for supernodes, it is interesting to find out whether such a node can outperform a collection of separate nodes with an equal total number of transmitters and receivers. The focus is on the design, operation and performance of a supernode. Specifically, a single supernode $\mathcal{S}$ is considered, which is surrounded by many conventional nodes, each of which carries a small fraction of the network traffic. The goal is to increase the throughput of $\mathcal{S}$. (Since $\mathcal{S}$ is assumed to constitute a throughput bottleneck, maximizing its throughput also maximizes the network throughput.) Also, for any given inbound throughput, it is desirable to maximize the efficiency of channel usage, which is defined to be the reciprocal of the number of times a packet must be transmitted until it is received successfully by its destination. While much research has, is and still needs to be done to further the understanding of spread-spectrum channels and CDMA, (e.g. [24, 48, 49],) the emphasis here is on identifying and understanding issues which, while being inherent to this type of channel, are valid regardless of the exact channel characteristics.

In section 2, we present a model for packet reception in the CDMA environment. Since the most common use of spread-spectrum channels is currently in packet-radio networks [52], it is initially assumed that a node cannot receive and transmit concurrently. In section 3, the Slotted ALOHA [53] multi-access scheme is assumed, with packet lengths of exactly one slot. This permits an evaluation of multiple receivers, multiple transmitters and a combination thereof, while keeping the analysis simple. Since packet radio networks are frequently not fully connected, the neighborhood of $\mathcal{S}$ is modeled as a two-hop topology.

Viewing a network as a graph whose nodes correspond to network nodes, there is a link from node i to node j with tag k if and only if node j can hear transmissions of node i and has a receiver for code k. In networks employing CDMA with Receiver-Directed Codes (CDMA/RDC), whereby nodes are allocated disjoint sets of codes for reception, each transmission activates only one link. It is therefore possible to *mask individual links* of the graph. This is different from narrowband networks, in which the decision as to whether or not ever activate links is made jointly for all the outgoing links of a node. As a digression from the issue of multiple transmitters and receivers, but in line with the performance-enhancement of a busy node, the slotted model is used in section 4, in conjunction with a single-hop topology, for the exploration of link masking as a means of increasing $\mathcal{S}$'s inbound throughput. Link masking is particularly relevant to such slotted systems.

While the slotted model is convenient for analysis, and is sufficient for exposing a number of issues, it nevertheless hides the effect of time-capture. This is, of course, particularly relevant to the reception of packets. In section 5, an unslotted model is used to further study the design and operation of $\mathcal{S}$'s receivers, focusing on time capture and on the resulting design tradeoffs. Several multi-receiver node architectures and code-assignment policies are proposed and compared.

Although spread-spectrum and CDMA are currently used primarily in packet-radio networks, they can also be used with local-area networks over low-attenuation media, in which transmitted and received signal levels are similar. In such networks, a node can receive while transmitting, and the half-duplex restriction is no longer required. This may also be true of packet radio networks in which the receiver and transmitter of a node are not collocated, or at least use separate antennas. In section 6, the results for half-duplex nodes are adapted to the case of full-duplex nodes. Section 7 concludes the chapter.

4.2 Model for Packet-Reception

A packet consists of two fields: (i) *preamble* of fixed length, and (ii) *data*. The reception of a packet consists of two phases: (i) synchronization onto the preamble, and (ii) reception of the data portion. In the spread-spectrum environment, occasional contamination of the received signal is possible. Therefore, the data portion of the packet is sometimes encoded prior to the transmission in a way that permits correction of errors to a certain degree and detection of (some) uncorrected errors. The encoded bit sequence is obviously longer than the original one. The process of encoding the data and later decoding it and correcting errors is referred to as *forward error correction*, or FEC [54]. Upon completion of its reception, a received packet is decoded by the recipient. If it is error-free, the reception is considered successful; otherwise, the packet is rejected. Packets that are not received successfully are lost and must be retransmitted at a later time. We will distinguish between *raw throughput*, consisting of all received packets, and *error-free throughput*, consisting only of those packets that are received successfully, i.e., free of errors.

The synchronization phase is successful if and only if (i) the receiving node is not transmitting, (ii) the arriving packet is *receivable*, i.e., its preamble does not overlap (at S) with that of another packet that was transmitted on the same code, and (iii) there is an available receiver on the appropriate code. The phenomenon of

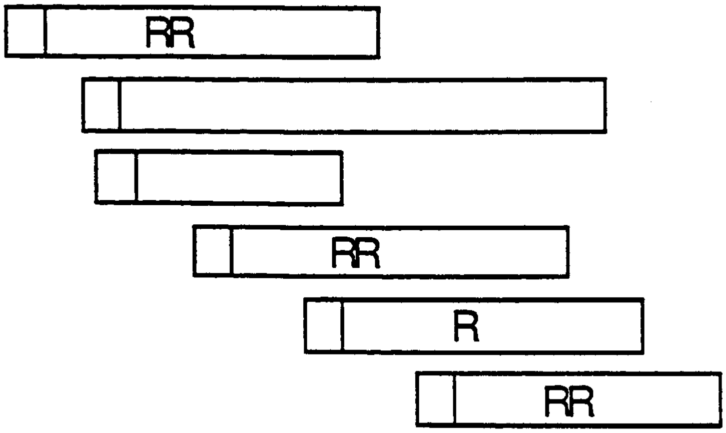


Fig. 4.1 Example of a packet arrival process. We assume that all packets are on a common code, and that there are two receivers. packets marked "RR" are received; those marked "R" are receivable but cannot find an available receiver, and the unmarked ones are nonreceivable.

overlapping preambles of packets with the same code will be referred to as *intracode interference.* Fig. 4.1 shows an example of a packet-arrival stream. The packets marked "RR" are receivable and find a receiver, so they are received. Those marked "R" are receivable but do not find a receiver. The unmarked ones are nonreceivable. In the figure, it is assumed that there are two receivers and a single code. An unsuccessful attempt to synchronize onto a nonreceivable packet does not prevent the receiver from synchronizing onto the next receivable packet, since the preamble of a receivable packet never overlaps with that of a nonreceivable one.

The finite capacity of the channel results in interference which depends primarily on the number of ongoing transmissions and is independent of code; this will be referred to as *intercode interference.* Intercode interference is assumed to manifest itself only in the form of erroneous bits in received packets, thus rendering those receptions unsuccessful; it cannot cause a receiver to abort an ongoing reception. From this, along with an assumption that a node does not begin transmitting when engaged in data-reception, it follows that the data-reception phase begins upon successful completion of the synchronization phase, and is always completed.

With the above model, throughput analysis can be carried out in two stages. Initially, the raw throughput is computed. This stage accounts for the loss of packets due to preamble-overlap on the same code and due to the lack of an available receiver. Both of these depend on the architecture, on the code assignment policy and on the level of S's inbound traffic, but not on channel parameters such as coding scheme, signal to noise ratio and capacity, or on the level of background traffic. The second stage accounts for the remaining cause for loss of packets, namely erroneous bits due to intercode interference, which depends only on the total traffic level and on channel parameters, and yields the error-free throughput. This approach decouples the architecture-dependent factors from the channel-dependent ones, thus permitting the raw-throughput results to be used in conjunction with intercode-interference results which are obtained for different coding schemes, levels of background traffic, etc.

The above model is an approximate one. The following paragraphs give some insight into the approximations and the consequences of using them.

Synchronization. In practice, the synchronization pattern is repeated several times in the preamble. Therefore, partial overlap of preambles with the same code may still permit synchronization onto them, and the model used here is thus somewhat pessimistic. Given a specific preamble design, our model can be used in a more accurate way by replacing the true preamble length with an appropriately shorter one. This has the desired effect of reducing the probability of preamble overlap for any given arrival rate. Another approximation involves the implicit assumption that inter-code interference does not affect the synchronization. The logic behind this approximation is that if the level of inter-code interference is such that a short, robust preamble is interfered with in a significant way, the probability of no errors in a received packet is very low, and such operating conditions are thus of very little interest.

Decoupling of error-freedom from reception. Let us consider the probability that a packet is error-free; i.e., the probability that if there were an infinite number of receivers, as well as some magic way of guaranteeing synchronization, the packet would be received successfully. Due to the Markovian nature of the system, the only dependence of this probability on the history of the system is through the number of ongoing transmissions at the time of arrival of the packet. Using k to denote this number, this probability is given by

$$P[\text{error-free}] = \sum_{m=0}^{\infty} P[k=m] \cdot P[\text{error-free}|m]. \tag{4.1}$$

Also, the probability that a packet is received and is error-free (with a finite number of receivers) can always be expressed as

$$P[\text{successful reception}] = P[\text{received}] \cdot P[\text{error-free}|\text{received}]. \tag{4.2}$$

The first term on the right hand side of (4.2) is equal to the ratio of the raw throughput to the mean packet-arrival rate, and the second one can be expressed as

$$P[\text{error-free}|\text{received}] = \sum_{m=0}^{\infty} P[k=m|\text{received}] \cdot P[\text{error-free}|m]. \tag{4.3}$$

Therefore, the decoupling approximation is close if and only if the knowledge that a packet was received has little effect on the distribution of the number of ongoing transmissions at the time of the packet's arrival. Furthermore, a very good first-order correction can be obtained by using the value of P[error-free] which corresponds to the correct value of the mean number of ongoing transmissions. This will be evaluated for a specific case and further elaborated upon in a later section.

4.3 Multiple Receivers and Multiple transmitters with Slotted ALOHA

With packet lengths of exactly one slot and all transmissions starting at the beginning of a slot, there cannot be partial overlap of transmissions. As a result, there is no time capture, and each receiver must have a different code. It also follows that a receivable packet is guaranteed to find an available receiver. Therefore, a packet is received if and only if the receiving node is not transmitting and hears no other transmission on the same code. Intercode interference manifests itself in the form of some function, $P_S(l)$, which is the probability that a packet that is received in the presence of $(l-1)$ other ongoing transmissions is found to be error-free.

4.3.1 Network Model

We consider a single supernode, $\mathcal{S}$, which is equipped with T transmitters and M receivers; (each receiver is assigned a different code;) $\mathcal{S}$ is surrounded by N conventional nodes (neighbors), and each neighbor is within range of Q other neighbors. There are also other, *external* nodes, which are not within range of $\mathcal{S}$ but are within range of some of its N neighbors. Fig. 4.2 shows an example with $N = 12,\ Q = 4$.

Each neighbor transmits independently according to a Bernoulli (p) process. A neighbor's transmission uses any given supernode code with probability $\frac{\alpha}{M}$, the code of any given neighbor (from among the Q that are within range) with probability $(1-\alpha)\cdot\frac{\beta}{Q}$, and the code of some external node with probability $(1-\alpha)\cdot(1-\beta)$. Each neighbor hears k transmissions of external nodes with probability $P_{ET}(k)$; the probability that such a transmission uses that neighbor's code is denoted by δ. (α, β, δ and $P_{ET}(k)$ are assumed to be given; in practice, they are a byproduct of the traffic pattern and the routing strategy.) The supernode $\mathcal{S}$ transmits in any given time slot with probability p_0. Two forms of synchronization between the

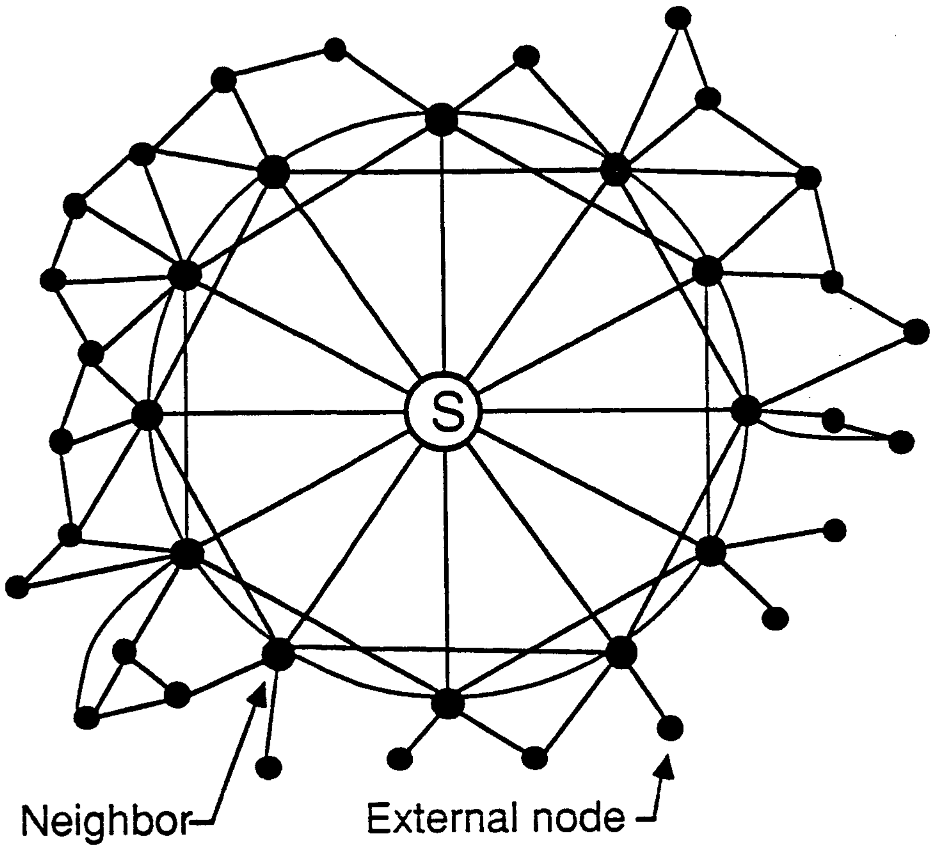


Fig. 4.2 Example of a topology for the analysis of throughput with slotted ALOHA. $N = 12$; $Q = 4$.

T supernode transmitters are enforced: ***code synchronization***, whereby concurrent transmissions by the supernode's transmitters employ different codes, and ***time synchronization***, whereby the T transmitters are either all idle or all transmitting. Without time synchronization, S would hardly ever be available for reception. Due to the relative simplicity of the slotted model, the error-free throughput will be calculated directly.

4.3.2 Multiple Receivers

For the calculation of S's inbound throughput, the external nodes need not be considered. Let $P_a(l)$ denote the probability that exactly one transmission uses a specific supernode code, given that there are l concurrent transmissions, and let $P_b(l)$ denote the probability that exactly l neighbors are transmitting in a given slot. The two probabilities are given by

$$P_a(l) = \frac{l\alpha}{M} \cdot \left(1 - \frac{\alpha}{M}\right)^{l-1}, \tag{4.4}$$

$$P_b(l) = \binom{N}{l} \cdot p^l \cdot (1-p)^{N-l}, \tag{4.5}$$

and the mean inbound throughput of each of $\mathcal{S}$'s receivers is

$$S_{in_i} = (1 - p_0) \cdot \sum_{l=1}^{N} P_a(l) \cdot P_b(l) \cdot P_S(l). \tag{4.6}$$

The average throughput into the supernode is therefore

$$S_{in} = M \cdot S_{in_i} = (1 - p_0) \sum_{l=1}^{N} \alpha \cdot l \cdot P_S(l) \binom{N}{l} p^l (1-p)^{N-l} \left(1 - \frac{\alpha}{M}\right)^{l-1}. \tag{4.7}$$

For the simple case wherein all transmissions of the neighboring nodes are intended for $\mathcal{S}$, and $P_S(l) = 1$ for $0 \leq l \leq L$ and 0 otherwise, the dependence of p_{opt} (the value of p that maximizes S_{in}) on M and L is shown in Fig. 4.3, and Fig. 4.4 shows $S_{in}/(1-p_0)$ versus M (maximized over p). The figures show that inbound throughput is initially proportional to the number of supernode receivers (and codes), since *intracode* interference is the limiting factor, but the marginal benefit eventually tapers off due to *intercode* interference. Although the optimal number of receivers is infinite, a practical maximum would be a number beyond which the benefit of additional receivers is small.

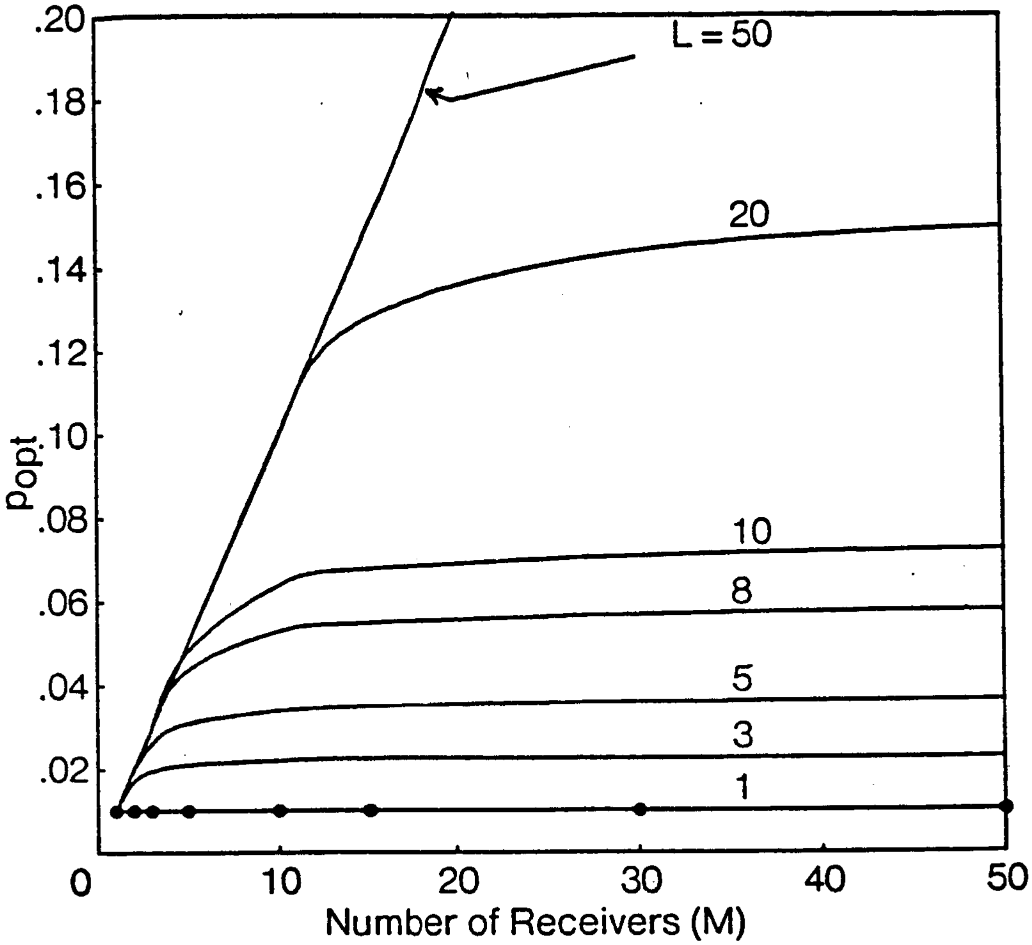


Fig. 4.3 Multiple receivers: neighbor's optimal probability of transmission. $N = 100$.

4.3.3 Multiple Transmitters

Initially, the probability of successful reception of a supernode packet is calculated. To do so, one must take into account S, the intended recipient, the Q neighbors that are within range of the intended recipient, and the external nodes. Given that the intended recipient is not transmitting, and that it hears exactly $q \leq Q$ transmissions of other neighbors of S along with T transmissions by S and

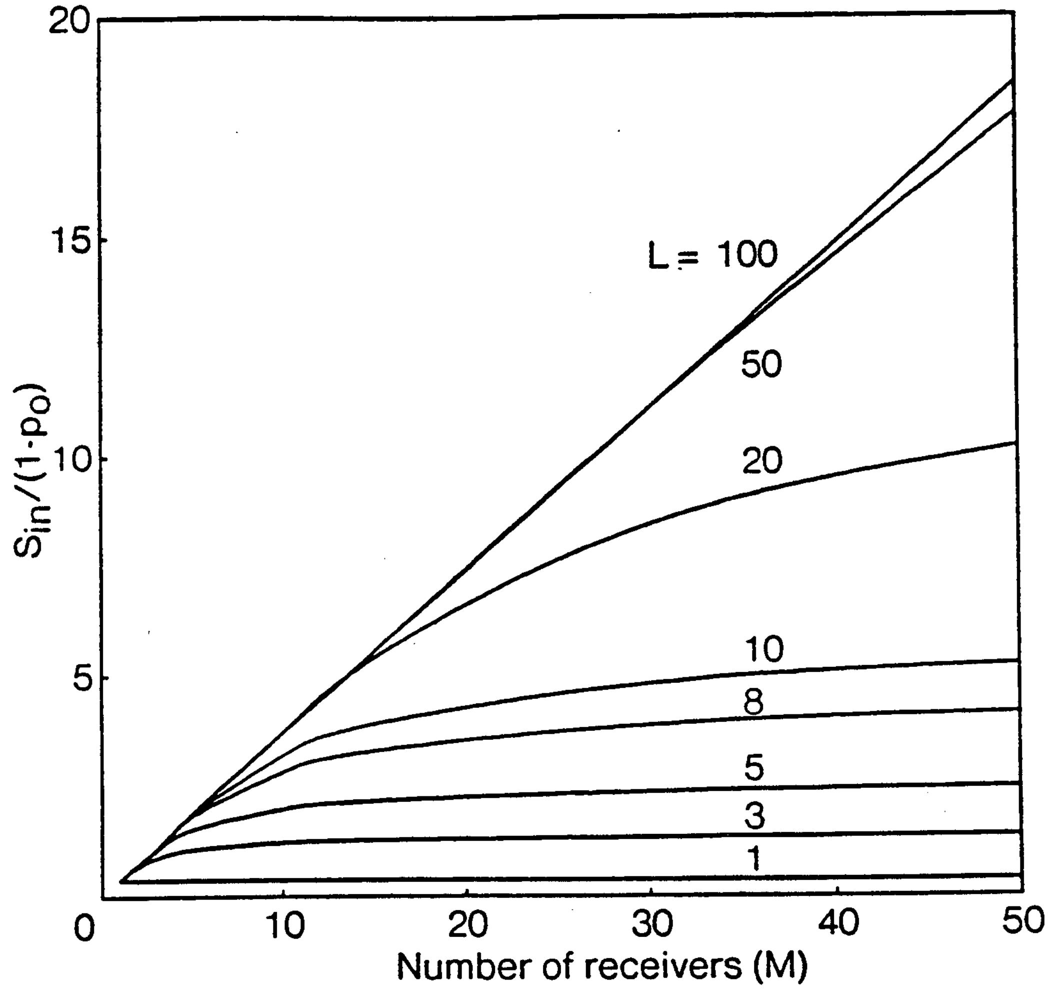


Fig. 4.4 Maximum normalized inbound throughput with multiple receivers. (Slotted). $N = 100$; $p = p_{opt}(N, M, L)$.

k transmissions of external nodes:

$$P[\text{no } \textit{intracode} \text{ interference}|q,k,T] = \left[1 - \beta \cdot \frac{1-\alpha}{Q}\right]^q \cdot (1-\delta)^k, \qquad (4.8)$$

$$P[\text{no } \textit{intercode} \text{ interference}|q,k,T] = P_S(q+k+T). \qquad (4.9)$$

Given the number of transmissions by each type of nodes (S; neighbors; external nodes), *intercode* and *intracode* interference are independent of each other. Consequently:

$$\begin{aligned} &P[\text{ reception of a given supernode packet}|q,k,T] \\ &= (1-p)\left[\left[1-\beta\frac{1-\alpha}{N-1}\right]^q (1-\delta)^k\right] P_S(q+k+T). \end{aligned} \tag{4.10}$$

Since the transmissions of $\mathcal{S}$, its neighbors and external nodes are independent of each other, the probabilities of the conditions can be calculated separately. In fact, the only one that really needs to be calculated is the probability of q:

$$P[q \text{ of the } Q \text{ relevant neighbors of } \mathcal{S} \text{ transmit}] = \binom{Q}{q} \cdot p^q \cdot (1-p)^{Q-q}; 0 \leq q \leq Q. \tag{4.11}$$

Finally, relaxing all the conditions except for the T supernode transmissions, the outbound throughput is given by:

$$\begin{aligned} S_{out} &= T \cdot p_0 \cdot (1-p) \cdot \\ &\cdot \sum_{q=0}^{Q} \sum_{k=0}^{\infty} \left\{ \left[\left(1-\beta\frac{1-\alpha}{Q}\right)^q (1-\delta)^k \right] P_S(q+k+T) \left[\binom{Q}{q} p^q (1-p)^{Q-q} P_{ET}(k) \right] \right\}. \end{aligned} \tag{4.12}$$

To illustrates the various trends, we now turn to the simple case of a fully connected network consisting of $\mathcal{S}$ and its N neighbors, in which all transmissions by the neighbors use $\mathcal{S}$'s codes. Therefore: $Q = N-1$; $\beta = 1$; $P_{ET}(k) = 0\ \forall k \neq 0$; $\alpha = 1$. $P_S(l)$ is approximated by a step function: $P_S(l) = 1$ for $0 \leq l \leq L$ and 0 otherwise. In this case, (4.12) reduces to

$$S_{out} = p_0 \cdot T \cdot (1-p) \cdot \sum_{q=0}^{\min\{L-T,N-1\}} \binom{N-1}{q} p^q (1-p)^{N-1-q} \tag{4.13}$$

From (4.13) it can be seen that there is a nontrivial value of T which maximizes S_{out}: a very small number of transmitters results in low throughput due to the

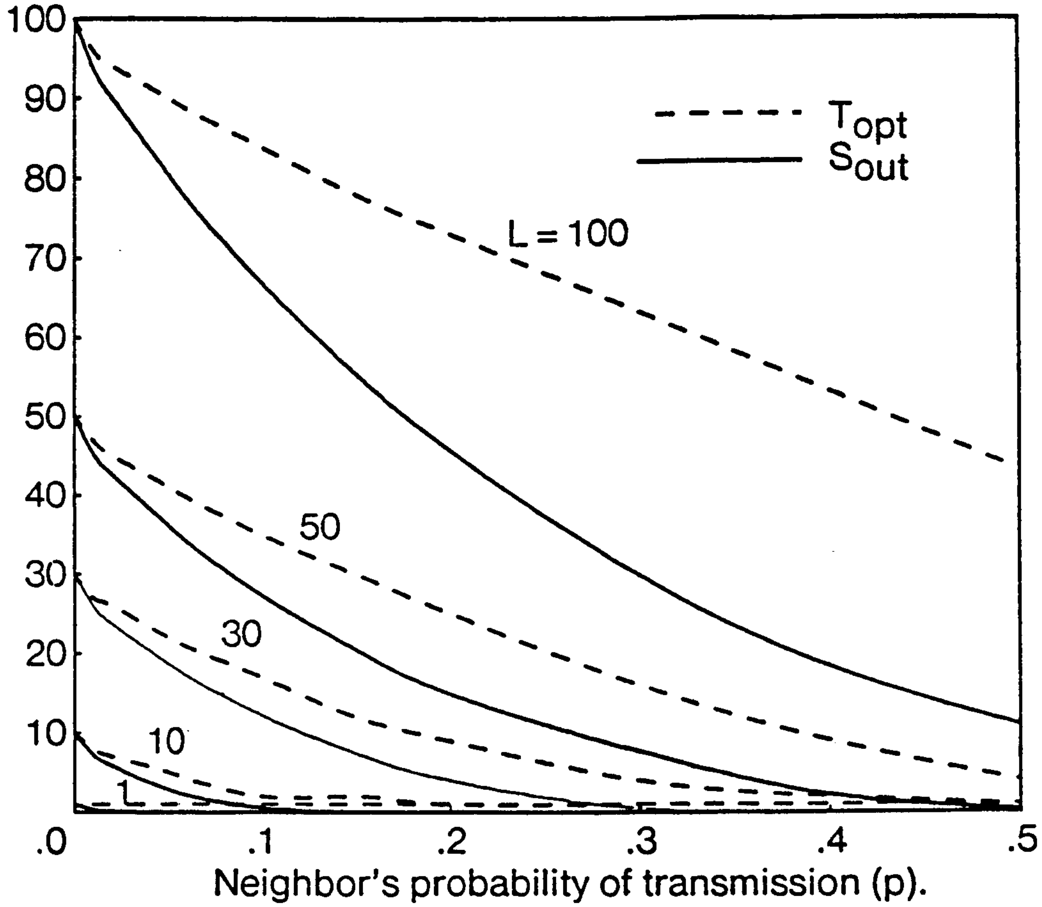


Fig. 4.5 Optimal number of supernode transmitters and maximum normalized outbound throughput. (Slotted.) $N = 100$.

fact that each transmitter can transmit at most one packet per slot. Increasing the number of transmitter increases the transmission rate, but increases probability of destructive intercode interference, thereby decreasing the probability of reception of any given packet. Fig. 4.5 presents the results for the specific case. The dashed curves show the dependence of T_{opt} (the value of T that maximizes S_{out}) on p and L; T_{opt} is independent of p_0. The solid curves show the dependence of S_{out}/p_0 on p

and L with $T = T_{opt}$. Note that the throughput per transmitter also decreases with an increase in p, even if the optimal number of transmitters is used. Although the results would vary with N, the primary dependence is on $p \cdot N$, which indicates the fraction of the channel capacity that is unavailable for S's transmissions. It is also interesting to observe that the throughput per transmitter increases as we decrease T (and everything else remains unchanged). Consequently:

$$S_{out}(T) \geq \frac{T}{T_{opt}} \cdot S_{out}(T_{opt}), \quad T < T_{opt} \tag{4.14}$$

The opposite is true for $T > T_{opt}$.

4.3.4 Multiple Transmitters and Receivers

In sections 3.2 and 3.3, the goal was to maximize throughput in one direction, assuming that the parameters associated with the other direction had been set and are thus part of the environment. The two directions are now combined in order to address the problem of maximizing S_{in} and S_{out} subject to the constraint $S_{in}/S_{out} = \gamma$. In its most general form, this is a multidimensional optimization over the parameters M, T, p and p_0, (N and L are given). It will be assumed that M is also given, since the unconstrained maximization of throughput occurs with $M = \infty$. One could also formulate several related problems. For example, there may be a cost constraint that determines $(M + T)$, and the goal will be to find (M, T, p, p_0) that maximize throughput.

Since $S_{in_{max}}(p_0)$ decreases as p_0 increases, and since setting $T = T_{opt}(p)$ minimizes the value of p_0 required to achieve any given value of S_{out}, yet does not affect $S_{in}/(1 - p_0)$, (because once S is transmitting it cannot receive, regardless of T,) it follows that T should best be set to $T_{opt}(p)$, as computed earlier. The problem thus reduces to a maximization of S_{in} over p, such that $S_{in}/S_{out} = \gamma$. Fig. 4.6 depicts

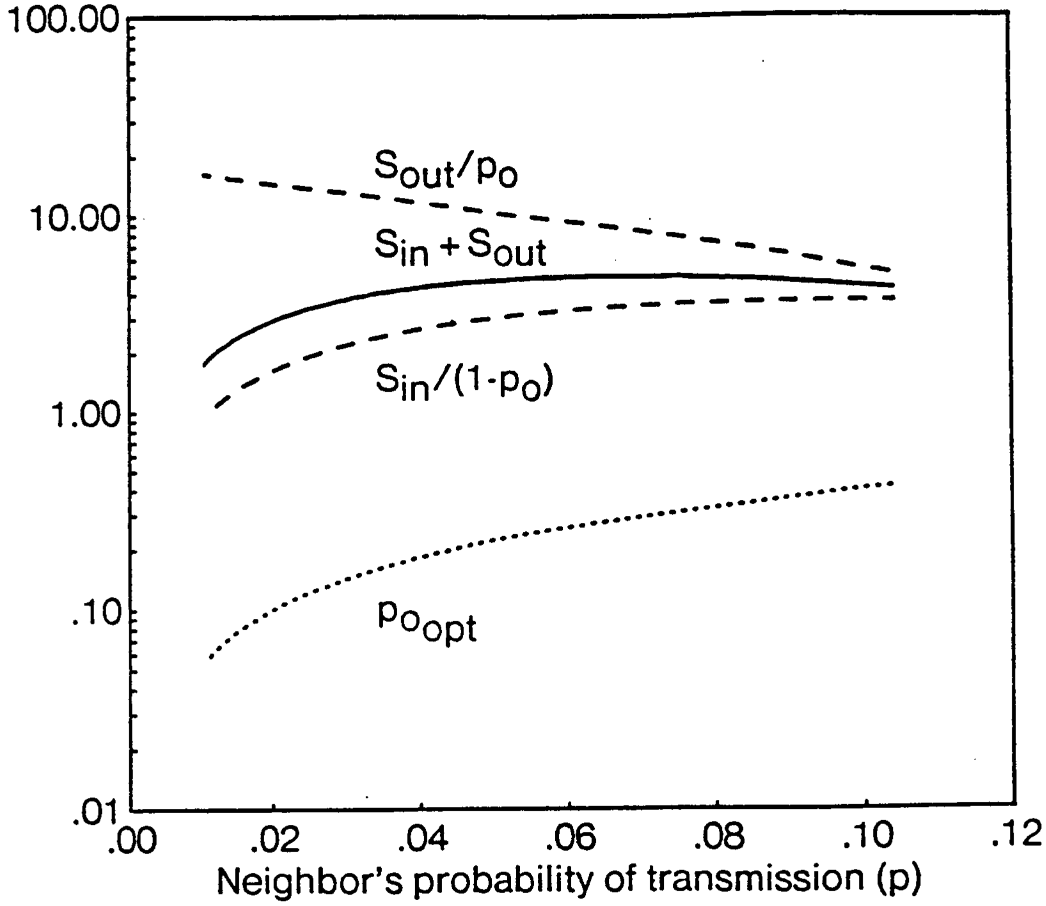


Fig. 4.6 Multiple receivers and transmitters: $S_{in} = S_{out}$; $N = 100$; $L = 20$; $M = 10$; $0 < p < p_{opt}(N, M, L,\text{inbound only})$.

$S_{in}/(1-p_0), S_{out}/p_0, p_0$, and $(S_{in} + S_{out})_{max}$ versus p for $\gamma = 1$. Fig. 4.7 depicts $p_{opt}, p_{0_{opt}}$ and $(S_{in} + S_{out})_{max}$ versus γ. Both figures were generated for the specific case that was used in sections 3.2 and 3.3.

The design problem becomes much more complicated when there are several supernodes, since transmissions intended for one supernode can be interfered with by transmissions intended for other supernodes as well as by transmissions of other

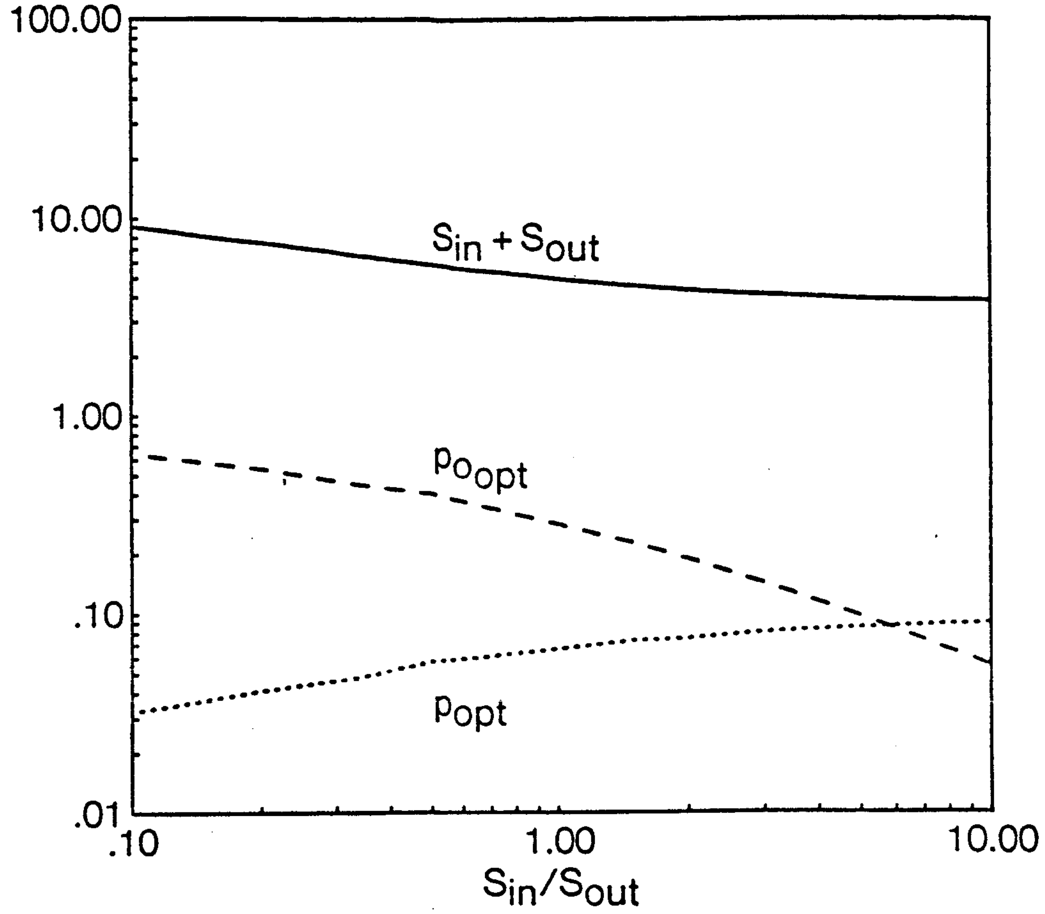


Fig. 4.7 Multiple receivers and transmitters: maximum throughput versus γ. $N = 100$; $L = 20$; $M = 10$.

supernodes. Consequently, S_{in} does depend on the number of transmitters used by the supernodes, and the optimal number of supernode transmitters is no longer the one obtained in section C. In fact, it is smaller. Although the optimization is multidimensional and, as a result, more complicated, the computation for each choice of parameters is similar to the simple case.

4.4 Link Masking

In this section, it is shown how link masking can be used to increase S's inbound throughput by reducing the size of the contending population. Specifically, the funneling of all the inbound traffic destined for any given receiver of S through a subset of S's neighbors (***authorized neighbors*** for that receiver) is considered, thus masking S's remaining inbound links. Recalling that the throughput of a conventional Slotted ALOHA channel is $1/e$ for $N = \infty$ and 0.5 for $N = 2$, link masking can be expected to increase inbound throughput by up to 36%. To prevent obstruction of the main issue at hand, we set $M = 1$ and $L = \infty$. The accommodation of multiple receivers is straightforward, provided that $N \geq 2M$; otherwise, it is slightly more complicated due to an overlap of the funnels for different receivers. The accommodation of finite values of L is similar to the previous sections, and the relationship between raw and error-free throughput will be commented upon.

The analysis of link masking as applied to this specific example is similar to that of routing packets to a central node in a narrowband network via a sequence of repeaters [55,56,57]. In the referenced studies, it was assumed that the central node never transmits. Here, it is studied for CDMA/RDC, allowing $p_0 \geq 0$.

Let us define the *routing graph* to be the directed graph consisting of the union of the paths to be used to route packets from each node to S, but excluding the initial hop of those paths. The goal is to determine the maximum attainable throughput into S and the simplest routing graph that can achieve it. (Shortest paths in terms of hops.) Since throughput with slotted ALOHA increases as the size of the contending population decreases, each node in the simplest routing graph should transmit to exactly one other node. Combining this with the requirement that all paths of the routing graph end at S, it follows that the simplest routing graph is a tree which has S as its root. Before proceeding, some notation must be introduced:

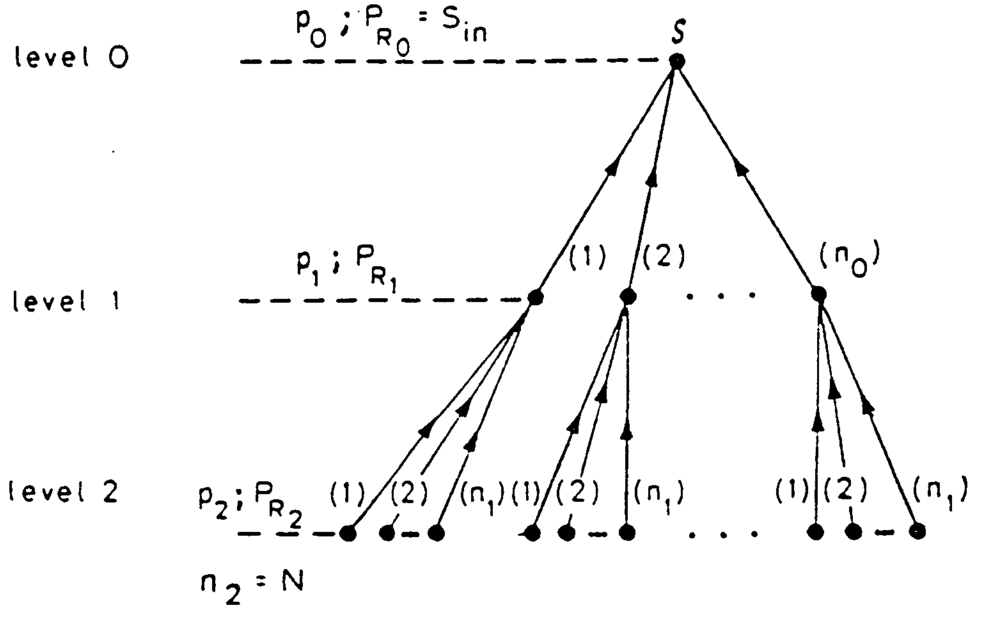


Fig. 4.8 Link masking: typical routing tree.

level– The distance in hops from a node to S via the routing tree.

n_i – The number of level-$(i+1)$ nodes that may transmit to each level-i node.

p_i – Probability of transmission of any given level-i node in any given slot.

P_{R_i} – Probability that any given level-i node receives a packet in any given slot.

A typical routing tree is shown in Fig. 4.8. The routing tree is governed by the following set of equations:

$$P_{R_i} = n_i p_{i+1}(1 - p_{i+1})^{n_i - 1}(1 - p_i) \qquad i = 0.1, ... \qquad (4.15)$$

$$P_{R_{i+1}} = \frac{P_{R_i}}{n_i} \qquad (4.16)$$

Note that, in a narrowband network, assuming a symmetric hearing matrix that represents a tree, the right hand side of (4.15) would contain an additional factor of $(1 - p_{i-1})$, representing transmissions by the father of the level-i node [56].

Assuming that the upper bound of $0.5(1-p_0)$ on $\mathcal{S}$'s inbound throughput is achievable, we proceed to impose requirements on the routing tree, beginning with level 0, and obtain upper bounds on n_i along with matching values of p_i. Obtaining $n_i \leq \infty$ is the indicator for having reached the leaves of the minimal routing tree.

level 0: $n_0 = 2$; $p_1 = 0.5$;

level 1: combining this with (4.15) and (4.16), and noting that, for any value of n_i, P_{R_i} is maximized by setting $p_{i+1} = \frac{1}{n_i}$, yields:

$$\left(1 - \frac{1}{n_1}\right)^{n_1 - 1} = 0.5 \cdot (1 - p_0) \tag{4.17}$$

Solving (4.17) for $\lfloor n_1 \rfloor$ yields the upper bound on n_1 as a function of p_0: for $p_0 = 0$, $n_1 = 2$; for $p_0 \geq 0.264$ $(= 1 - 2/e)$, $n_1 \leq \infty$.

level 2: similarly, and recalling that $p_i \leq 0.5$, the upper bound on n_2 is ∞, regardless of p_0.

The two extremes of the required routing tree are shown in Fig. 4.9. The size of the routing tree is independent of the number of network nodes. In the remainder of this section, the focus here is on link masking using a height-1 binary tree ($n_0 = 2$, $n_1 \leq \infty$, 2-hop paths); the 1st hop is from the source to an authorized neighbor, and the 2nd hop is from an authorized neighbor to $\mathcal{S}$. It will be shown that 2-hop link masking comes close to achieving $S_{in} = 0.5(1-p_0)$, thus rendering more complicated routing graphs, such as the height-2 binary tree (3-hop link masking), unnecessary.

When $p_0 < 0.264$, the 1st hop cannot support the maximal throughput of the 2nd hop (with $p_1 = 0.5$). For this case, $S_{in_{max}}(p_0)$ is calculated by substituting $n_0 = 2$ in (4.15) and in (4.16), replacing P_{R_1} in (4.16) with a tight lower bound of $\frac{1}{e}(1-p_1)$, and solving (4.15) and (4.16) for S_{in}. The lower bound is tight (it is

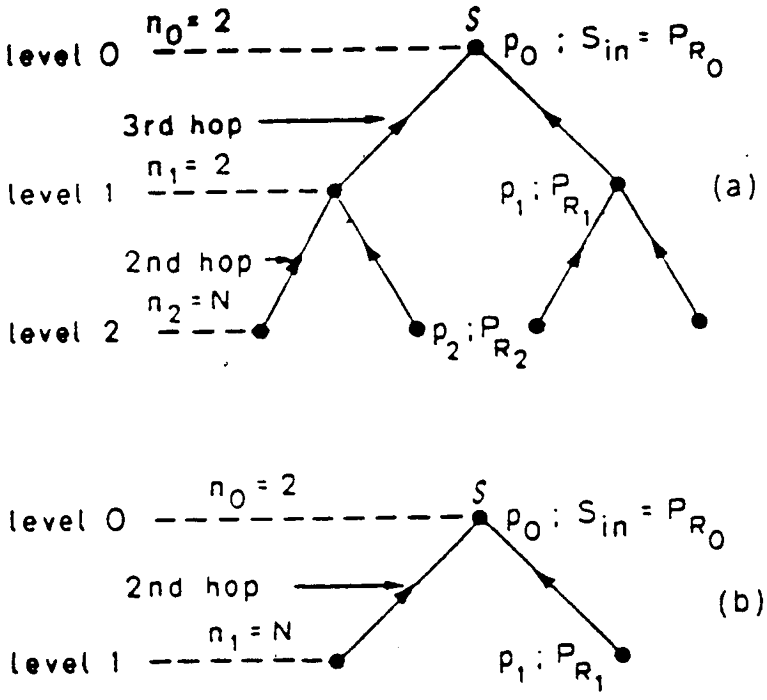


Fig. 4.9 Link masking: minimal routing trees. a) $p_0 < (1-2/e)$; b) $p_0 \geq (1-2/e)$.

exact for $n_1 = N = \infty$); the maximization is over p_1. When $0.264 \leq p_0 < 1$, the bottleneck is in the 2nd hop and $S_{in_{max}} = 0.5(1-p_0)$. In summary:

$$S_{in_{max}}(p_0) = \begin{cases} \frac{2}{e} \cdot \left(1 - \frac{e^{-1}}{1-p_0}\right), & 0 \leq p_0 \leq 0.264 \\ 0.5(1-p_0), & 0.264 < p_0 < 1 \end{cases} \tag{4.18}$$

A plot of $S_{in_{max}}/(1-p_0)$ versus p_0 is shown in Fig. 4.10; results for direct transmissions and for 3-hop link masking are presented for reference.

Having demonstrated the throughput advantage of 2-hop link masking over direct transmissions, their performance in terms of efficiency is next compared. Although it seems that the use of 2-hop link masking reduces the efficiency by a factor of two, since each packet must be transmitted at least twice, it will be shown that this is not always the case.

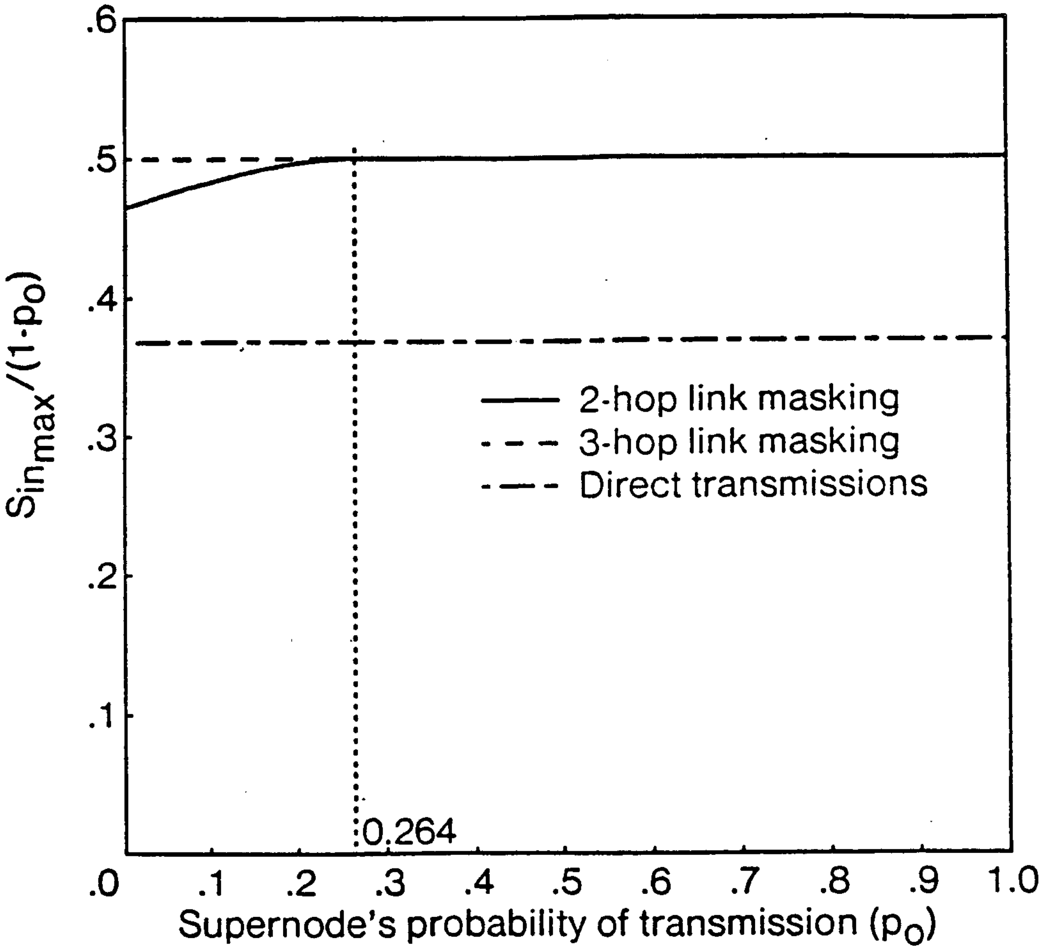


Fig. 4.10 Link masking: maximum normalized inbound throughput. (Slotted.) $M = 1$; $N, L >> M$.

Direct transmission. Let us set $p = \frac{\theta}{N}$. Assuming large N:

$$\theta \cdot e^{-\theta} = \frac{S_{in}}{(1-p_0)} \qquad 0 \leq S_{in} \leq \frac{1}{e} \cdot (1-p_0);\ 0 < \theta \leq 1 \tag{4.19}$$

and the transmission rate is θ.

2-hop link masking. The transmission rates on the 1st and 2nd hops are $N \cdot p_2$ and

$2 \cdot p_1$, respectively. Solving (4.15) for p_1:

$$p_1 = 0.5 \cdot \left(1 - \sqrt{1 - 2\frac{S_{in}}{1 - p_0}}\right) \tag{4.20}$$

Let us set $p_2 = \frac{2\theta'}{N}$, so that each level-1 node sees a population of N nodes, each of which transmits to it with probability $\frac{\theta'}{N}$. Substituting the latter for p_{i+1} and $\frac{S_{in}}{2}$ for P_{R_1} in (4.15), and assuming large N yields

$$\theta' e^{-\theta'} = \frac{0.5S_{in}}{(1 - p_1)}, \tag{4.21}$$

and the transmission rate is $2(\theta' + p_1)$. (The feasible (p_0, S_{in}) combinations for equations (4.19),(4.20) and (4.21) are represented by the regions under the appropriate curves in Fig. 4.10.) Fig. 4.11 depicts $S_{in_{max}}$ for 2-hop link masking, along with the aggregate transmission rate on each of the hops, as a function of p_0. For $p_0 < 1 - \frac{2}{e}$, the throughput bottleneck is seen to be in the 1st hop; consequently, the transmission rate on the 2nd hop is lower than that which maximizes the throughput of that hop. For $p_0 > 1 - \frac{2}{e}$, the bottleneck is on the 2nd hop and the transmission rate on the 1st hop is lower than the one which maximizes its throughput. Fig. 4.12 depicts the efficiency of channel usage as function of inbound throughput for various values of p_0; curves are presented for direct transmissions and for 2-hop link masking. Both S_{in} and the efficiency are divided by $(1 - p_0)$ in order to remove the effect of S's unavailability for reception due to its own transmission. It can be seen that the efficiency of direct transmissions depends on p_0 only through the factor of $(1 - p_0)$. Therefore, having divided by the factor, the curves are independent of p_0. With 2-hop link masking, however, there is an additional trend, namely an increase in efficiency with an increase in p_0. To understand this, note that increasing p_0 while keeping p (direct transmissions) and p_1 (2-hop link masking) unchanged has the same effect on the throughput of the two schemes, and the efficiency of the 2nd

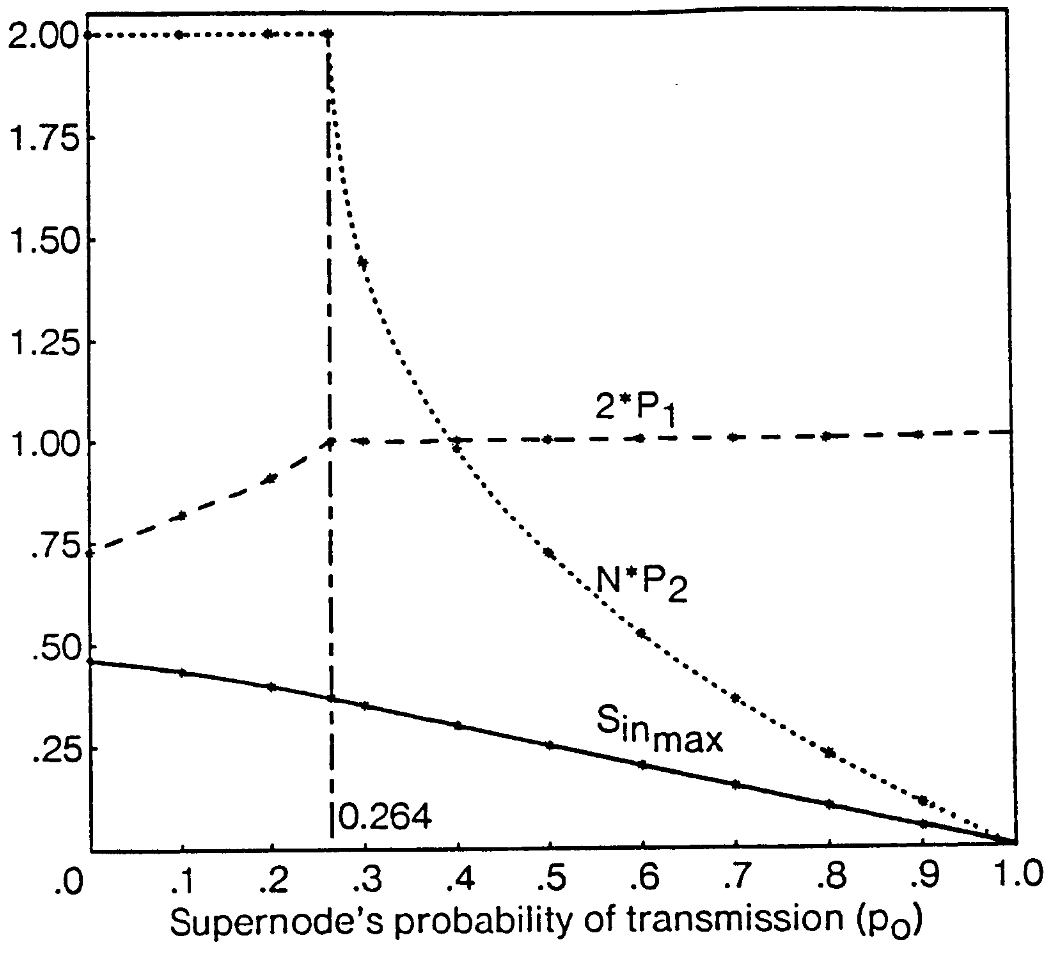


Fig. 4.11 Maximal inbound throughput and required transmission rates with 2-hop link masking. $M = 1$; $N, L >> M$.

hop (link masking) relative to that of direct transmissions is also unchanged (since the probability of reception of a packet is proportional to $(1 - p_0)$ in both cases). However, due to the drop in S_{in}, P_{R_1} (link masking) also decreases, and the value of p_2 required to achieve it is lower. This, in turn, results in increased efficiency of the 1st hop. Consequently, as p_0 is increased, there is an overall improvement in the efficiency of link masking relative to that of direct transmissions. Fig. 4.13

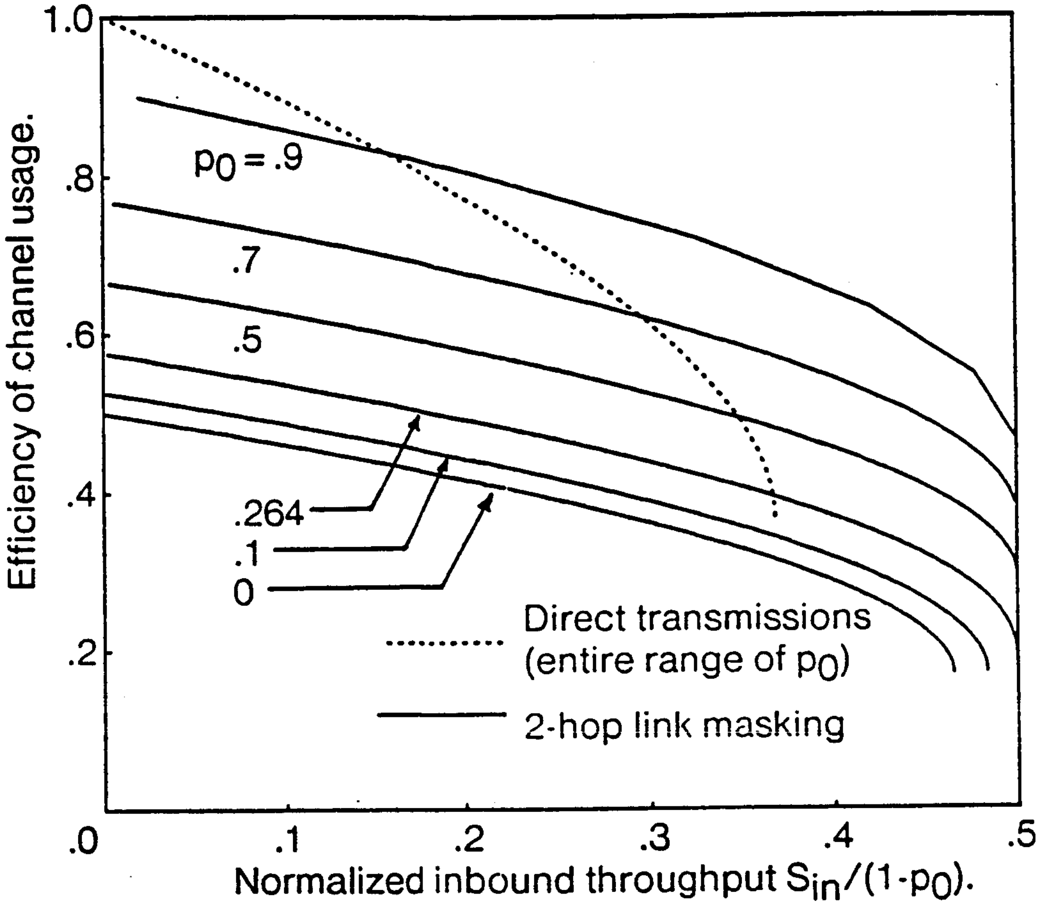


Fig. 4.12 Efficiency of channel usage with 2-hop link masking and with direct transmissions.

depicts $S_{in_{max}}(p_0)$ as a function of p_0, for direct transmissions and for 2-hop link masking. The feasible (p_0, S_{in}) combinations for each scheme-are represented by the region under the appropriate curve. The dashed curve in that figure is the equal efficiency line. Below it, direct transmissions are more efficient (fewer transmissions per reception) than 2-hop link masking. Above the boundary, 2-hop link masking is more efficient. Observe that, for low values of p_0, direct transmissions are more efficient as long as they are feasible, and the boundary is thus simply $S_{in_{max}(p_0)}$ for

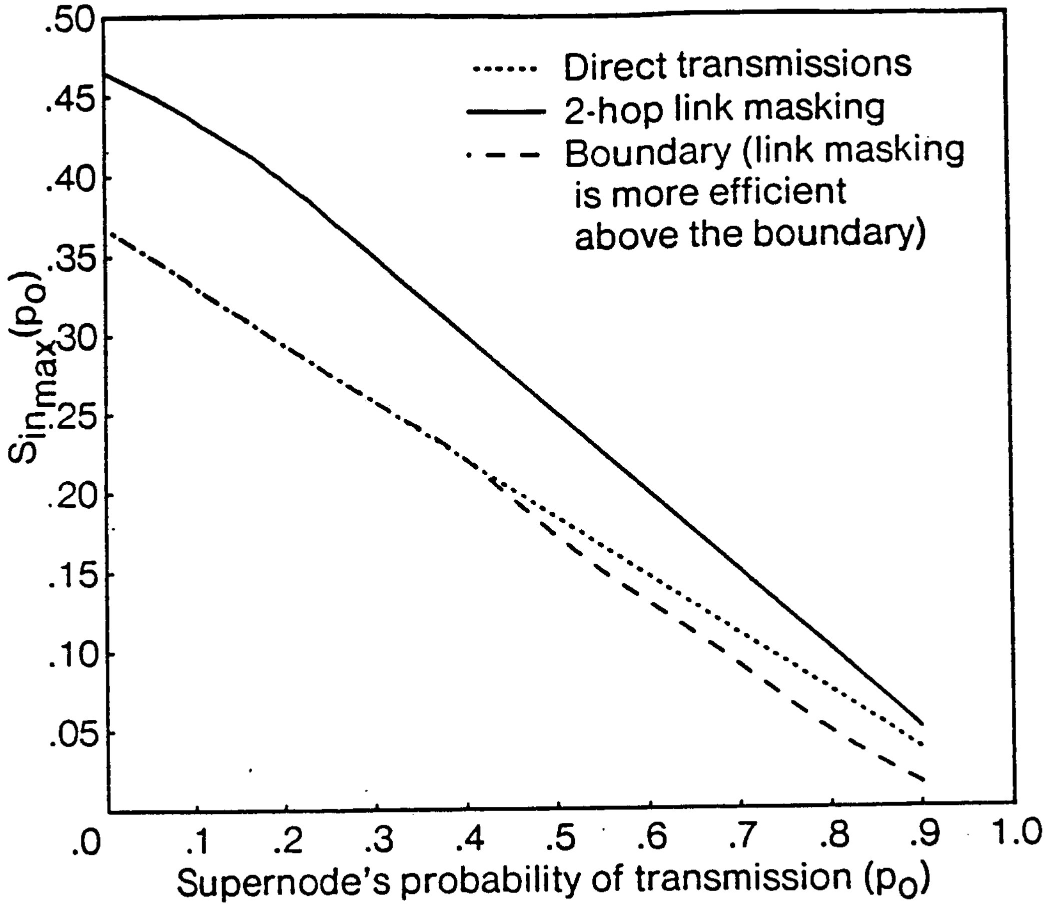


Fig. 4.13 Feasibility and superiority boundaries for direct transmissions and for 2-hop link masking. $M = 1$; $N, L >> M$.

direct transmissions. However, as p_0 increases, the boundary moves into the feasible domain of direct transmissions.

The results of this section depend on N only when it becomes small. The indicator for the closeness of the approximation in assuming "very large N" is the relative difference between $(1 - 1/N)^{N-1}$ and $1/e$. For example, the differences for $N = 5, 10, 20$ are 11%, 5.3% and 2.5%, respectively.

As seen in Fig. 4.11, the aggregate transmission rate associated with a single receiver that employs link masking is at most 3, which is typically much smaller than channel capacity. Therefore, if only few receivers mask their inbound links and the level of background traffic is low, the raw throughput is a good approximation for the error-free throughput; in such a case, the effect of intercode interference on the performance of link masking relative to that of direct transmissions is also very small. However, intercode interference does limit the number of receivers that may employ link masking efficiently; the limit depends on channel capacity and on the level of background traffic. (another upper limit is $\frac{1}{3}$ of the number of nodes.)

The protocol required to support 2-hop link masking is very simple and robust: each network node keeps two addresses for $\mathcal{S}$, which are actually the addresses of the two authorized neighbors, and uses either one (at random). This has the additional benefit of balancing the load on the two authorized neighbors.

Link masking should not be used for outbound single-destination traffic, because the probability of reception of a supernode packet by its lightly-loaded destination node is higher than the probability of reception by the busy forwarding node. It may, however, be beneficial for multi-destination packets. $\mathcal{S}$ would transmit such a packet once to one of its neighbors, which will retransmit it on the code of each of the intended recipients. A similar approach can be taken with acknowledgments for inbound traffic. Since acknowledgments are very short, $\mathcal{S}$ can collect several acknowledgments into a single packet and send them to one of its neighbors, which would then distribute them to their destinations. Unlike the inbound funnel, which consisted of two specific neighbors, the outbound funnel can change dynamically. ($\mathcal{S}$ may select an ad-hoc "helper" for each such acknowledgment packet.)

4.5 Multiple Receivers with Time-Capture

In this section, an unslotted model is considered, which exposes the effect of time capture. The preambles of packets are of fixed length, and the length of the data portion follows an exponential distribution. Without loss of generality, the transmission time of a preamble is selected as the unit of time, and the mean transmission time of the data portion is denoted by $\frac{1}{\mu}$.

4.5.1 Network Model (Unslotted)

Again, a single supernode $\mathcal{S}$ is considered; it is equipped with M receivers and is allocated $N_c \leq M$ codes for reception. The arrival process of packets with any given code, consisting of new as well as retransmitted packets, is Poisson, and is i.i.d. from code to code. The aggregate arrival rate at $\mathcal{S}$ is λ. The length of arriving packets is assumed to be i.i.d. according to the aforementioned packet-length distribution. To avoid unnecessary complexity of the mathematical derivations, it is assumed that $\mathcal{S}$ never transmits. The accommodation of $\mathcal{S}$'s transmissions is deferred to the end of this section. We now proceed to calculate raw throughput for various architectures and to compare them. Unlike in the slotted case, the throughput analysis here is carried out in two stages. (Raw throughput and error-free throughput.)

4.5.2 Fixed-Code-Assignment Architecture (FCAA)

$\mathcal{S}$'s receivers are partitioned into groups of R[†]; all receivers of any given group are permanently assigned the same code. Each group has a controller, which designates one of the idle receivers (if any) to receive the next incoming packet. This

[†] Throughout the discussion, R is assumed to divide M. In practice, the number of receivers (M) is an upper bound on R; also, if the desired value of R does not divide M, groups of different sizes will have to be constructed. Lastly, note that when $R = 1$, no use is made of the fact that the receivers are collocated.

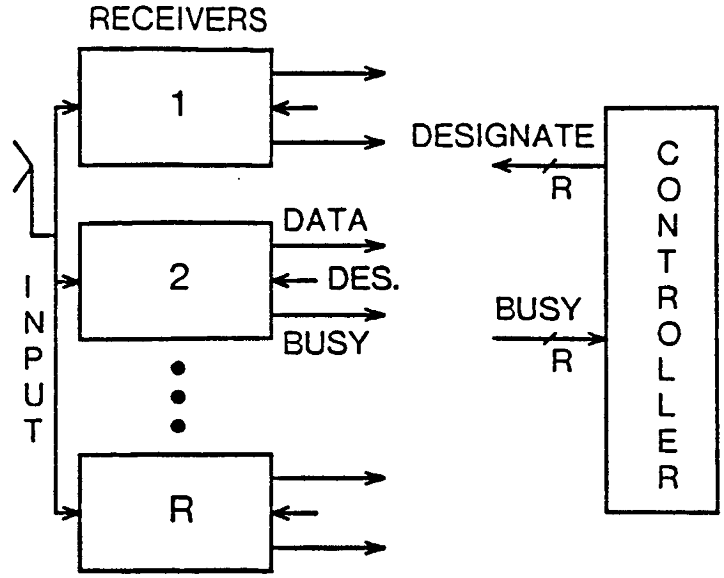


Fig. 4.14 Fixed code-assignment architecture. (FCAA.) A single group of R receivers is shown.

architecture is illustrated in Fig. 4.14. The packet arrival rate to a group is $\lambda' \triangleq \frac{R}{M}\lambda$. We next proceed to derive the raw throughput for FCAA.

Based on the model for packet reception, each group of receivers may be studied separately, considering only its R receivers and only arriving packets with its code. Furthermore, only receivable packets need to be considered. Intuitively, it seems that inbound throughput (S_{in}) should be maximized (over R and λ') by assigning each receiver a unique code ($R = 1$), since such an assignment minimizes the likelihood of intracode interference. We prove, by contradiction, that this is indeed the case: assume that S_{in} is maximized by setting $R = R_0$, $R_0 > 1$, and $\lambda = \lambda_0$. Consider now a second system, with $\lambda = R_0 \cdot \lambda_0$ and $R = 1$. The two systems have the same value of $\lambda' \triangleq \frac{\lambda \cdot R}{M}$; consequently, the arrival process of receivable packets to a group is the same in both systems. Since the ongoing reception of a receivable packet cannot be interfered with, it follows that, for any process of receivable-packet arrivals to the group, the throughput of any given receiver in the group is maximized if all the packets are directed to it (as opposed to being shared with other receivers). As a result, the inbound throughput of each receiver in the second system is higher than in the first one; this, in turn, results in a higher aggregate throughput and contradicts the optimality assumption. Note that this result is independent of packet length and of the number of receivers.

For any given inbound throughput S_{in}, the optimal group size R_{opt} is defined to be the value of R that maximizes efficiency. We claim that, for $S_{in} < S_{max}$, there are cases in which $R_{opt} > 1$. To see why this may indeed be the case, let us interpret R_{opt} as the value of R that maximizes S_{in} for a given value of λ. Observe that, for given values of λ and M, the probability of receivability is maximized if $R = 1$, because setting $R = 1$ minimizes the arrival rate of packets with any given code. On the other hand, such a choice minimizes resource-sharing and therefore maximizes the likelihood of a packet being discarded because all the receivers of its group are busy, even if there are idle receivers in other groups. There is hence a design tradeoff in selecting the value of R.

For fixed preamble length, (one,) the arrival process of receivable packets is nothing but the departure (reception) process of a Pure ALOHA [42] system with Poisson arrivals (rate λ'), zero capture and packets of unit length. These interdeparture times are i.i.d., and the mean rate of departures is $\lambda' e^{-2\lambda'}$. The Laplace transform of the probability density function of the interdeparture-time random variable X was derived by Takagi [58] as

$$X^*(s) = \frac{\lambda' e^{-(s+\lambda')}[s + \lambda' e^{-(s+\lambda')}]}{s^2 + s\lambda'[1 + e^{-(s+\lambda')}] + \lambda'^2 e^{-2(s+\lambda')}} \tag{4.22}$$

Let $\{Y(t),\ t \geq 0\}$ be the random process representing the number of busy receivers at time t. Next, let $\{Y(t_n)\}_{n=1}^{\infty}$ be the embedded process at $\{t_n\}_{n=1}^{\infty}$, where t_n represents the arrival time of the nth receivable packet. We now make a key observation, namely that interarrival times of receivable packets always exceed the preamble length. Since there is no queue, this implies that an arriving receivable packet always finds all the busy receivers in the data-reception phase, whose duration is exponentially distributed, having completed the fixed-duration synchronization phase. Consequently, $Y(t_n)$ constitutes a complete specification of the state of the system at $t = t_n$. Recalling that interarrival times of receivable packets are i.i.d, we conclude that $\{Y(t_n)\}_{n=1}^{\infty}$ is a Markov chain; we conveniently denote it by $\{Y_n\}_{n=1}^{\infty}$. Defining $\Pi = (\pi_0, \pi_1, \pi_2, ..., \pi_R)$ to be the steady-state probability vector, the probability of a receivable packet finding an available receiver is simply $(1 - \pi_R)$. It is independent of that packet's length.

To construct the transition probability matrix $P \triangleq [p_{ij}] \triangleq P\{Y_{n+1} = j | Y_n = i\}$, let us initially condition on $X \triangleq t_{n+1} - t_n = x$. Assuming that $Y_n = i$, $i < R$, i.e., assuming that the nth receivable packet found a receiver, $(i + 1 - j)$ receivers must complete service in time x in order to have $Y_{n+1} = j$. At the beginning of x, i of the $i + 1$ busy receivers are in the exponentially distributed data-reception phase, and the remaining one, which is receiving the nth receivable packet, is at the beginning of the fixed-duration synchronization phase. A slightly different situation occurs when $Y_n = R$: the nth receivable packet is lost, and the number of busy receivers stays R, each of which is in the exponentially distributed data-reception phase of its ongoing reception. The general expression for $p_{ij|X}(x)$ is given by:

$$p_{ij|X}(x) = \begin{cases} [1-e^{-\mu(x-1)}]\cdot\binom{i}{j}\cdot[1-e^{-\mu x}]^{i-j}\cdot e^{-\mu xj} & \\ +e^{-\mu(x-1)}\binom{i}{j-1}[1-e^{-\mu x}]^{(i+1)-j}\cdot e^{-\mu x(j-1)}, & 0 < j \leq i < R; \\ \binom{i}{j}\cdot[1-e^{-\mu x}]^{i-j}\cdot e^{-\mu xj}, & 0 \leq j \leq i = R; \\ [1-e^{-\mu(x-1)}]\cdot[1-e^{-\mu x}]^{i}, & j=0,\ i<R; \\ e^{-\mu(x-1)}\cdot e^{-\mu xi}, & j=i+1,\ i<R; \\ 0, & \text{otherwise.} \end{cases} \tag{4.23}$$

Relaxing the condition on X by letting

$$p_{ij} = \int_0^\infty f_X(x)\cdot p_{ij|X}(x)dx,$$

and recalling that $f_X(x) = 0$ for $x < 1$ yields:

$$p_{i,j} = \begin{cases} \binom{i}{j}\sum_{m=0}^{i-j}\left[(-1)^m\binom{i-j}{m}[X^*(\mu_{m+j}) - e^{\mu}X^*(\mu_{m+j+1})]\right] & \\ +e^{\mu}\binom{i}{j-1}\left[\sum_{m=0}^{i+1-j}\left[(-1)^m\binom{i+1-j}{m}X^*(\mu_{m+j})\right]\right], & 0 < j \leq i < R; \\ \binom{i}{j}\sum_{m=0}^{i-j}\left[(-1)^m\binom{i-j}{m}X^*(\mu_{m+j})\right], & j \leq i = R; \\ \sum_{m=0}^{i}\left[(-1)^m\binom{i}{m}[X^*(\mu_m) - e^{\mu}X^*(\mu_{m+1})]\right], & j=0,\ i<R; \\ e^{\mu}X^*(\mu_j), & 0 < j = i+1 \leq R; \\ 0, & \text{otherwise,} \end{cases} \tag{4.24}$$

where $\mu_k \triangleq k\mu$. Since $p_{i,j} = 0$ for $j > i+1$, the state-probability vector Π is given by the recursive expression

$$\pi_{i-1} = \frac{1}{p_{i-1,i}} \cdot \left[\pi_i - \sum_{m=i}^{R} \pi_m \cdot p_{m,i} \right], \quad i = 1, 2, ..., R \qquad (4.25)$$

along with the constraint that $\sum_{i=0}^{R} \pi_i = 1$.

Fig. 4.15 shows plots of the inbound throughput per receiver $\frac{S_{in}}{M}$ as a function of the rate of packet arrivals per receiver $\frac{\lambda}{M}$ for various group sizes R; throughput is expressed in packets per unit time, which has been taken to be the preamble length. Observe that, for any given packet length and arrival rate per receiver $\frac{\lambda}{M}$, there is an optimal group size; for low arrival rates, R_{opt} is large, since the throughput bottleneck is in finding an available receiver, and the increased resource sharing that is brought about by larger groups is important. As the arrival rate increases, preamble collisions become the limiting factor, and consequently R_{opt} decreases until it eventually becomes one. The dependence of R_{opt} on λ and M is only through $\frac{\lambda}{M}$. By comparing the two parts of Fig. 4.15, it is also evident that the advantage of using large groups, namely the increased sharing of resources, is more pronounced for long packets than for short ones. In fact, for $\frac{1}{\mu} < 10$, it is most practical to use $R = 1$ regardless of the arrival rate. Finally, note that for very low arrival rates, throughput is insensitive to group size.

Fig. 4.16 depicts efficiency as a function of inbound throughput per receiver, for various group sizes. These results are obtained directly from the throughput results. Observe, for example, that with $\frac{1}{\mu} = 20$ and $S_{in}/M = 0.033$, the efficiency can be increased by 50% by using $R = 3$ rather than $R = 1$.

4.5.3 Dynamic-Code-Assignment Architectures (DCAA)

In the dynamic-code-assignment architectures (DCAA), there is a controller that assigns codes to idle receivers and designates, for each code, (if possible,) an

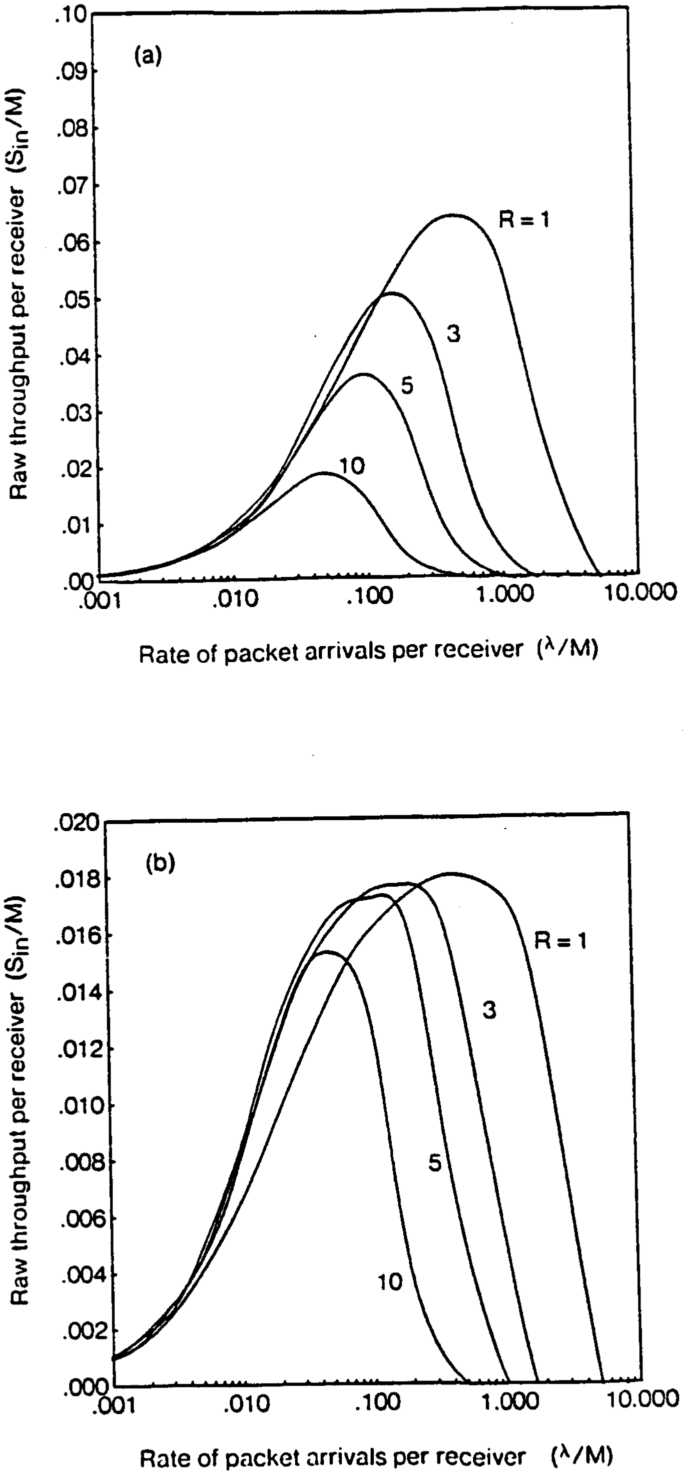


Fig. 4.15 Inbound throughput per receiver with FCAA for various group sizes. a) $\frac{1}{\mu} = 10$; b) $\frac{1}{\mu} = 50$.

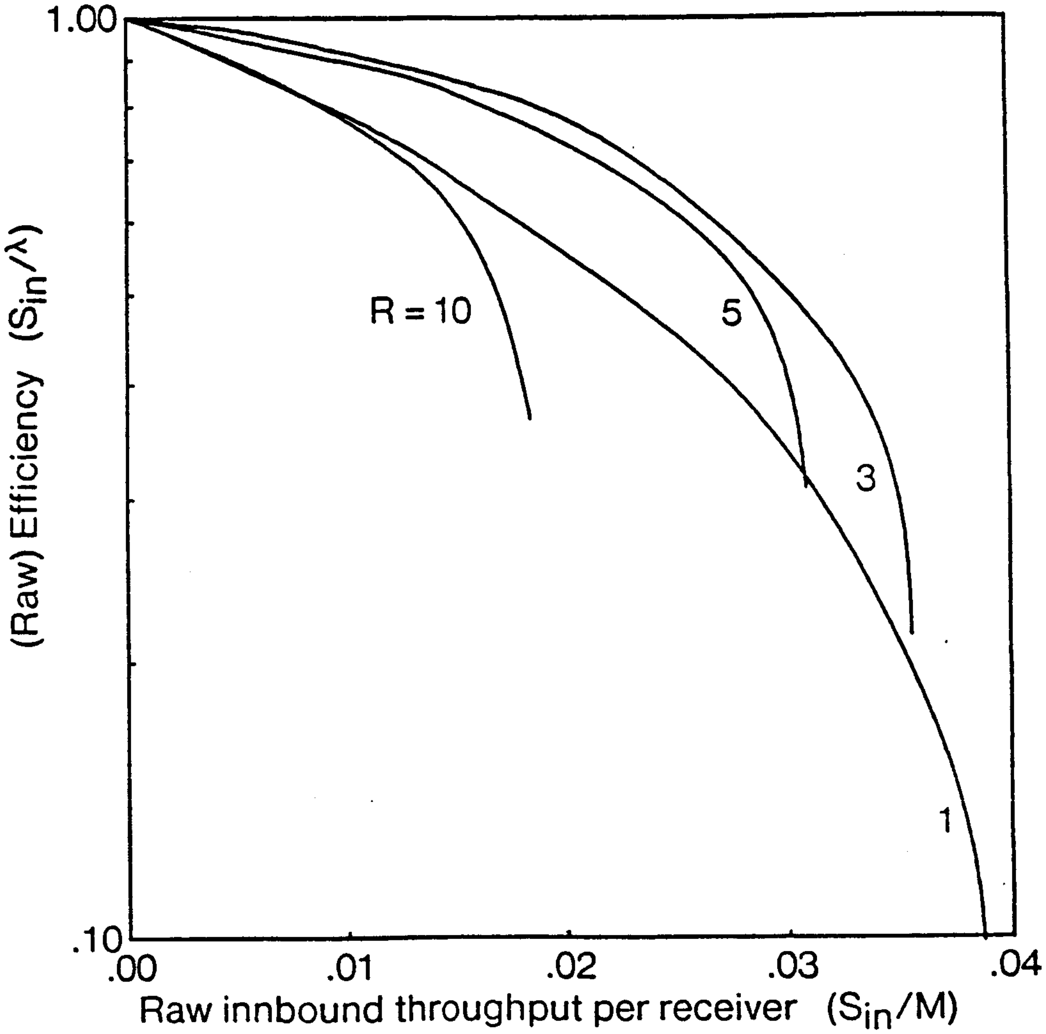


Fig. 4.16 Efficiency of channel usage with FCAA for various group sizes. $\frac{1}{\mu} = 20$.

idle receiver to await packets on that code. Fig. 4.17 shows a "generic" DCAA. The difference between various architectures of this class is in the knowledge that is available to the controller and in its consequent code-assignment policy. For any choice of (N_c, M, λ), the receivability with DCAA is the same as that with FCAA $(R = \frac{M}{N_c})$. The expected throughput enhancement stems from the increased sharing of receivers that is made possible by the dynamic reassignment of codes.

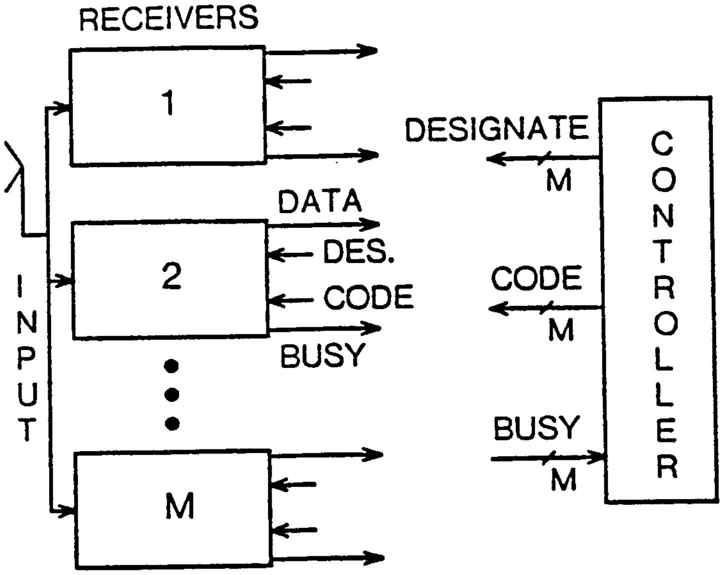


Fig. 4.17 Dynamic-code-assignment architectures. (DCAA.)

For example, if $M = 12$ and $N_c = 4$, at least nine receivers must be busy before a receivable packet may be dropped by a DCAA node, whereas with FCAA, if three receivers with the same code are busy and a receivable packet arrives on that code, it is lost. Two extreme variants of DCAA are now explored:

Random assignment (DCAA-RA)

The controller has no knowledge of the code of the next packet. Whenever the number of idle receivers, i, is smaller than N_c, the subset of codes that is covered is chosen at random. Therefore, a receivable packet that finds i idle receivers (receivers not busy in any phase of the reception of a receivable packet) is received with probability $\min\left\{\frac{i}{N_c}, 1\right\}$.

Optimal assignment (DCAA-OA)

The controller knows the code of the next arriving packet, and designates an idle receiver to attempt to receive it. Consequently, a receivable packet will not be lost so long as there is an idle receiver; a possible implementation of this seemingly

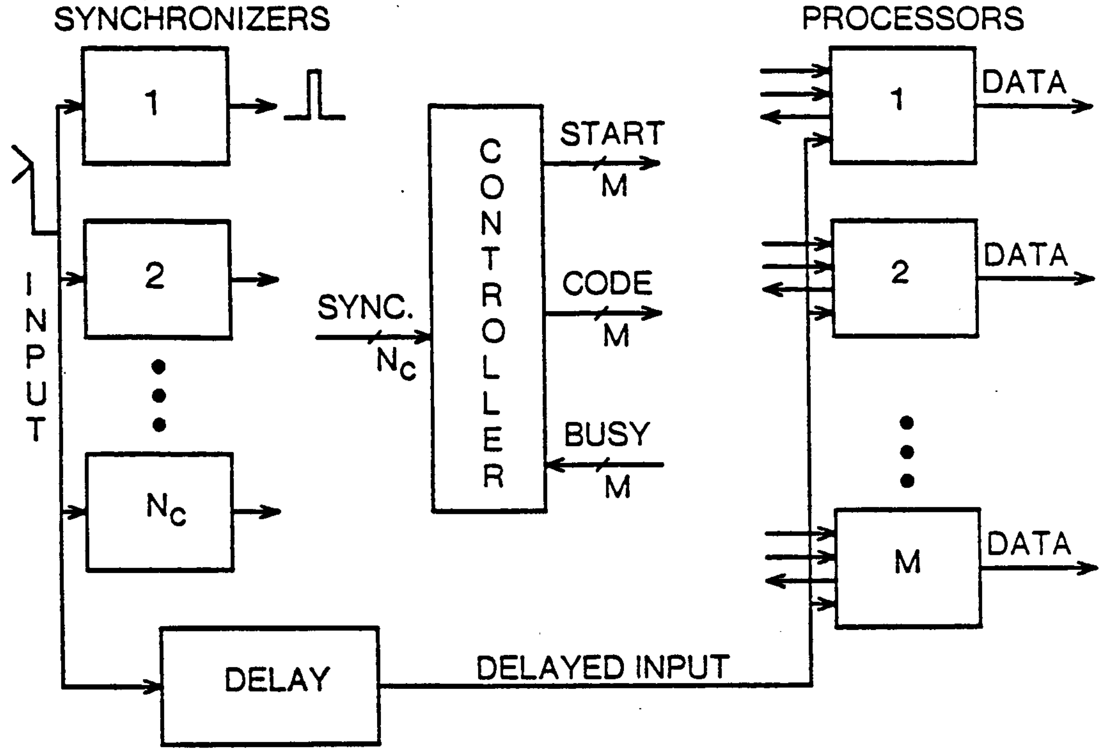


Fig. 4.18 Causal implementations of the seemingly noncausal DCAA-OA. The figure shows the pipelined version; the non pipelined version is obtained by removing the "start" signals and replacing "processors" with "receivers."

non-causal system is shown in Fig. 4.18. The multi-receiver node consists of a controller and of N_c synchronizers, followed by a pool of M "processors". At all times, the incoming signal appears at the input of each synchronizer and, after going through a delay line, at the input of each processor. Each synchronizer operates independently with a distinct code. Whenever it synchronizes onto a packet, it so notifies the controller and immediately resets itself to wait for a new packet. The controller then instructs one of the idle processors, if any, to process this packet using the appropriate code. This architecture is pipelined, so the processors only perform the data-reception phase. However, bit synchronization must be maintained between the synchronizer and the processor that cooperate in the reception of any given packet. Alternatively, the processors could be replaced by complete receivers, which would not rely on the synchronization performed by the synchronizer. (The only task of the synchronizer would be to supply the advance knowledge of the arriving packet's code.) This obviates the need for bit-synchronization between the synchronizer and the processor, but requires a longer delay and gives up the advantage of pipelining. We use the latter version in the analysis of DCAA-OA in order to avoid distortion of the comparison due to the throughput advantage of pipelining, which can also be used with the other schemes.

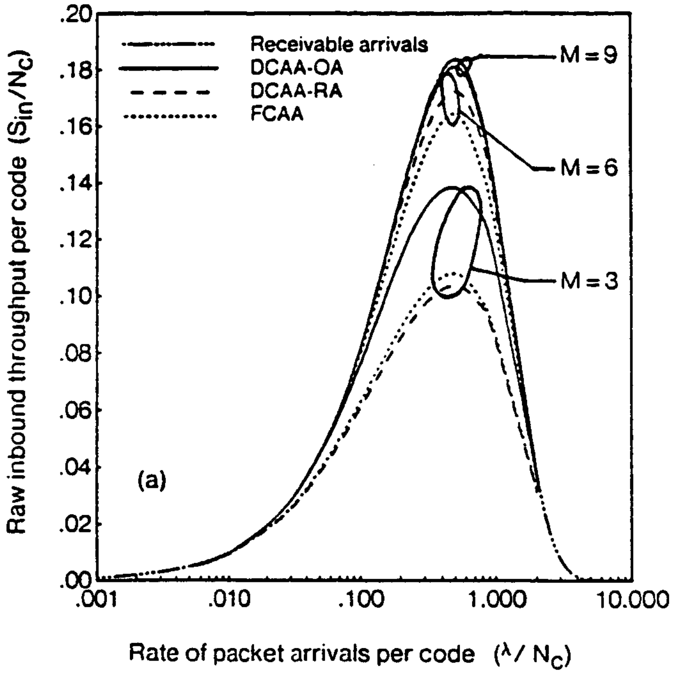
Analysis of DCAA-OA and of DCAA-RA differs from that of FCAA, because the interarrival times of receivable packets are not i.i.d. and the codes are not equiprobable and independent from arrival to arrival. We therefore resort to simulation.[†]

Fig. 4.19 depicts S_{in}/N_c versus λ/N_c for all three architectures; an additional curve shows the arrival rate of receivable packets per code, which is an upper bound on throughput. Note that the values are normalized per code, not per receiver. Curves are given for several values of M; the number of codes, N_c, is held fixed at 3. Graphs are presented for two packet lengths: a) $\frac{1}{\mu} = 3.3$ and b) $\frac{1}{\mu} = 10$. Fig. 4.20 depicts the efficiency of the three architectures with $\frac{1}{\mu} = 10$.

4.5.4 Performance Comparison

Schemes will be compared on the basis of throughput for equal arrival rates. Consequently, the same results also apply to efficiency. Comparing DCAA-RA with FCAA, FCAA appears to slightly outperform DCAA-RA when $M = N_c$, contrary to the expectation that they would be identical. This minor anomaly stems from

[†] In the simulation, we used a method known as "common random numbers" (the exact same interarrival times and packet lengths for the schemes under comparison). This decreases the variance of the relative results.



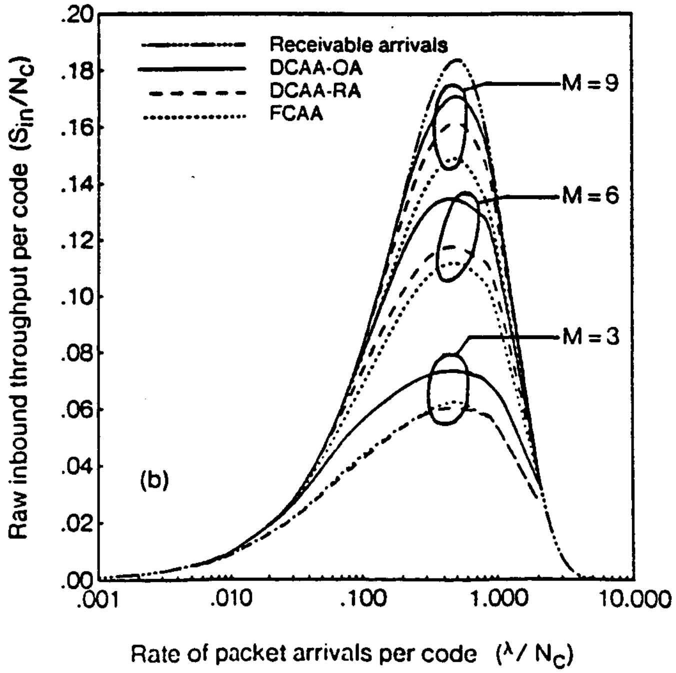


Fig. 4.19 Inbound throughput per code for FCAA, DCAA-RA and DCAA-OA. $N_c = 3$. a) $\frac{1}{\mu} = 3.33$; b) $\frac{1}{\mu} = 10$.

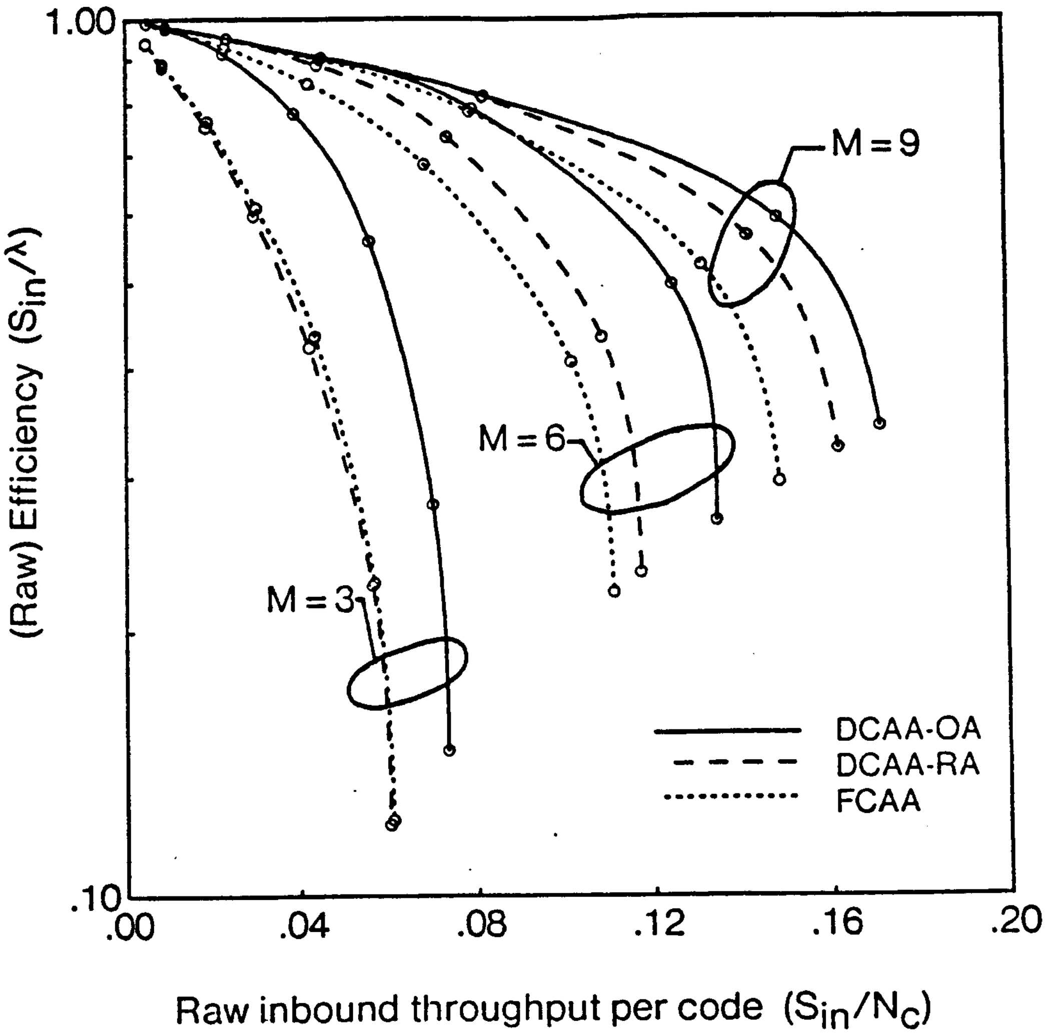


Fig. 4.20 Efficiency of channel usage with FCAA ($R = M/N_c$), DCAA-RA and DCAA-OA. $N_c = 3$; $\frac{1}{\mu} = 10$.

our modeling of DCAA-RA, and would disappear in a real implementation.[†] As the number of receivers increases, the advantage of DCAA-RA over FCAA becomes

[†] Our model for DCAA implies that there is a non-zero probability that, while one receiver is still performing the synchronization phase, another one is already designated to cover the same code. The latter is wasted during the remainder of the preamble, since no receivable packet can commence to arrive on that code at that time. In FCAA this can never happen when $M = N_c$. In practice, however, a designated receiver declares itself locked only at the end of the preamble, so no other receiver will be assigned to the same code until that time.

apparent.

Dynamic code assignment increases throughput as well as efficiency; the extent of the improvement depends heavily on the knowledge available to the controller. The advantage of DCAA over FCAA is most pronounced for intermediate values of λ: for very small values, the probability of reception is very high with any architecture; for very high values of λ, there is no improvement since receivability constitutes the bottleneck. Compared with FCAA, the maximum throughput is roughly 25% higher with DCAA-OA, and roughly 5% higher with DCAA-RA; this is for the range in which throughput is limited by receiver availability and not by receivability.

The dependence of the throughput advantage of DCAA on packet length is more complicated: consider, for example, the points of maximum throughput, and assume for the moment that the limiting factor is finding an available receiver. In this case, a receiver is nearly always designated to cover some code as soon as it becomes idle. The busy period of a receiver is therefore the packet length and its idle period is the time interval from the instant it becomes idle until a receivable packet arrives with the appropriate code. If the packet is long compared to the interarrival time of packets with any given code, (which, in turn, is on the order of preamble length,) the receiver's utilization is very high and cannot be improved much by decreasing the idle time through a knowledgeable code assignment as in DCAA-OA. If, on the other hand, the packet length is comparable to the preamble, (short packets,) there is more room for improvement. Consequently, one would expect a more significant improvement for short packets. There is, however, another trend: as the number of receivers is increased, the arrival rate of receivable packets becomes the limiting factor ("starved" receivers), in which case most codes are covered at any instant, thus causing the importance of knowledgeable code assignment to decrease.

Obviously, this happens first for short packets, since the busy periods of the receivers are shorter. One should therefore expect a more significant improvement for long packets in this case. Indeed, referring to Fig. 4.19 and comparing DCAA-OA with FCAA: for ($N_c = 3,\ M = 3$), the improvement is 32% for short packets ($\frac{1}{\mu} = 3.33$) and only 23% for longer ones ($\frac{1}{\mu} = 10$), whereas for ($N_c = 3,\ M = 9$) it is down to a mere 1% for the short ones, whereas for the long ones it is 7%.

4.5.5 Error-Free Throughput

Unlike the probability of reception of a packet, which is independent of its length, the probability of a packet being error-free may depend on its length. Consequently, the length-distribution of successfully received packets is not the same as that of transmitted or received packets, and stating traffic level or throughput in packets per unit time is ambiguous. The mean number of ongoing transmissions is therefore used as the new unit of traffic level; (replacing the mean packet-arrival rate;) raw throughput is redefined to be the mean number of ongoing receptions, and error-free throughput is the mean number of ongoing receptions of packets that will be found error-free. Traffic level and raw throughput, expressed in packets per unit time, can be converted to the new units by multiplying them by the mean packet length ($1 + \frac{1}{\mu}$).

Given the channel parameters, intercode interference can be characterized as the probability that a packet is error-free, (i.e., if it were acquired successfully and received, there would be no erroneous bits,) as a function of packet length and of the total traffic level λ_T. For a known length-distribution of received packets, it can be characterized simply by the probability that an arriving bit belongs to an error-free packet, as a function of λ_T. Then, given the level of background traffic, λ_{BG}, the error-free throughput for each value of $\mathcal{S}$'s inbound traffic level, λ_S, is the product of the raw throughput at λ_S and this probability at the corresponding value of λ_T. Strictly speaking, the distributions of packet lengths used in obtaining the channel characteristics and the raw throughput must be the same; however, the results are clearly insensitive to small differences.

The value of λ_T which corresponds to given values of λ_S and λ_{BG} is only approximately $(\lambda_S + \lambda_{BG})$, since knowing that a packet was received biases the distribution of $\mathcal{S}$'s inbound-traffic level at the time of the packet's arrival; furthermore, this bias depends on the architecture. Nevertheless, we claim that the bias is very small, except for the case of very short packets and very low traffic levels. (In this case, however, intercode interference is negligible altogether.) Our claim stems from the fact that the only knowledge gained from the fact that a packet with a preamble of length 1 is received at time t is that there was an available receiver on its code at time t and that no other packets with the same code commenced to arrive in $[t-1, t+1]$. No information is gained pertaining to background traffic or to arrivals after $t+1$. Very little information is gained pertaining to traffic on other codes (none in the case of FCAA), and not much regarding arrivals prior to $t-1$. Supported by simulation results,[†] we ignore the bias. (It should be emphasized that the only effect of this bias is to shift the curves horizontally, since it only affects λ_T. As will be seen in the curves that follow shortly, a horizontal shift of one curve with respect to the other by tens of percents has little effect on the relative performance of $\mathcal{S}$ with different code assignments. In other words, not only is the bias small, but the results are very insensitive to it.)

We now turn to a specific example. Fig. 4.21 , which is based on Fig. 4.6 in [50],

[†] We ran simulations for the case of FCAA with $M = 3$ and no background traffic; we used $R = 1, 3$ and $\frac{1}{\mu} = 10, 50$. For traffic levels of 10 ongoing transmissions or more, the bias of the traffic level is smaller than 10%. For levels of more than 30, the bias is smaller than 5% and for 75 it is down to 2%. The difference between the bias with $R = 1$ and with $R = 3$ is below 1%. These are the biases at the time of arrival; the bias decreases during the arrival of the remainder of the packet, so the average bias is even smaller.

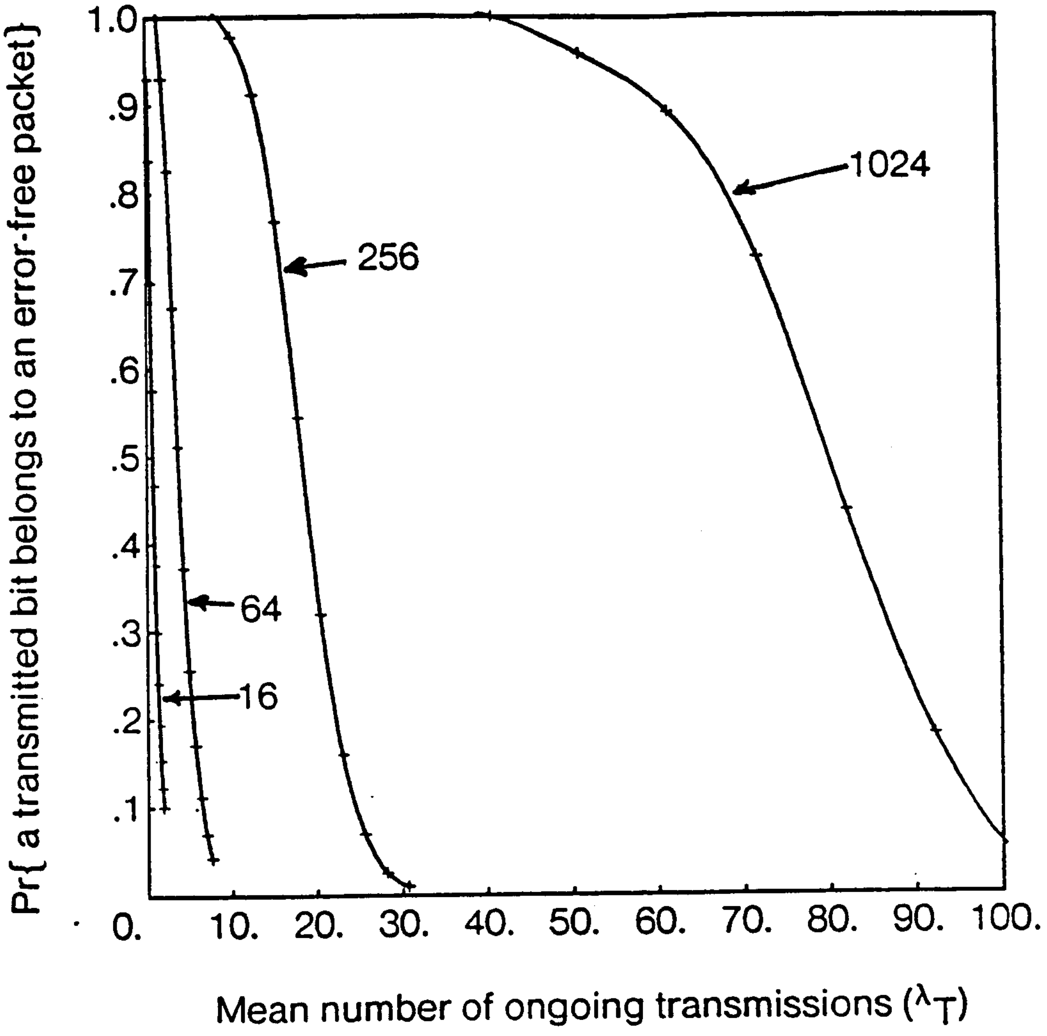


Fig. 4.21 Probability that a transmitted bit belongs to an error-free packet versus the mean number of ongoing transmissions. Packet-length is exponentially distributed with mean 1000 bits. Channel: DS-BPSK with convolutional FEC. The FEC contributes a factor of two to the number of chips per bit.[50]

shows the characteristics of a specific channel.† Plots are shown for 16, 64, 256

† DS-BPSK channel with FEC. PN codes are assumed to be sequences of jointly independent Bernoulli (1/2) random variables. FEC: convolutional coding with hard decision Viterbi decoding. The specific code used is the rate 1/2 constraint length 7 code with generator polynomial (in Octal) 171, 133. Packet lengths are exponentially distributed with mean 1000 bits, and signal to noise ratio is $E_b/N_0 = 8.0$.[50]

and 1024 chips per bit. (The FEC contributes a factor of 2 and the remainder is the PN spread-factor.)

Fig. 4.22 shows the error-free throughput for FCAA with $\frac{1}{\mu} = 50$ and $M = 3$, with no other traffic; this is a combination of Figs. 4.15(b) and 4.21. We see that, although maximum raw throughput is obtained with $R = 1$, error-free throughput is higher with $R = 3$; i.e., it is best to operate all three receivers on the same code. Including background traffic and using longer packets in the intercode interference model (a typical preamble length is 40 bits; with $\frac{1}{\mu} = 50$, the mean packet length should be 2000 bits instead of the 1000 used in [50]) would further increase the advantage of $R = 3$. The use of a more realistic preamble-collision model would have a similar effect.

Error-free throughput can be obtained for all architectures in the same manner. Intercode interference has no qualitative effect on the comparison between FCAA and DCAA, since the curves representing raw throughput never cross over. As for the optimal number of different codes to be used with a given number of receivers, that number is never smaller than the corresponding number for FCAA; for DCAA-OA, it is always best to use as many codes as possible, constrained only by code availability and by the budget for synchronizers.

The dependence of error-free throughput on the arrival rate can be summarized as follows: as the arrival rate increases, raw throughput increases until it begins to decrease due to preamble overlaps which render arriving packets nonreceivable. The probability that a received packet is error-free decreases as the arrival rate increases, but is initially very insensitive to arrival rate. At some point, however, this probability begins to fall off sharply. Error-free throughput is maximized (approximately) at the lower of two arrival rates: (i) the one which maximizes the raw throughput, and (ii) the one at which the probability of a packet being error-free

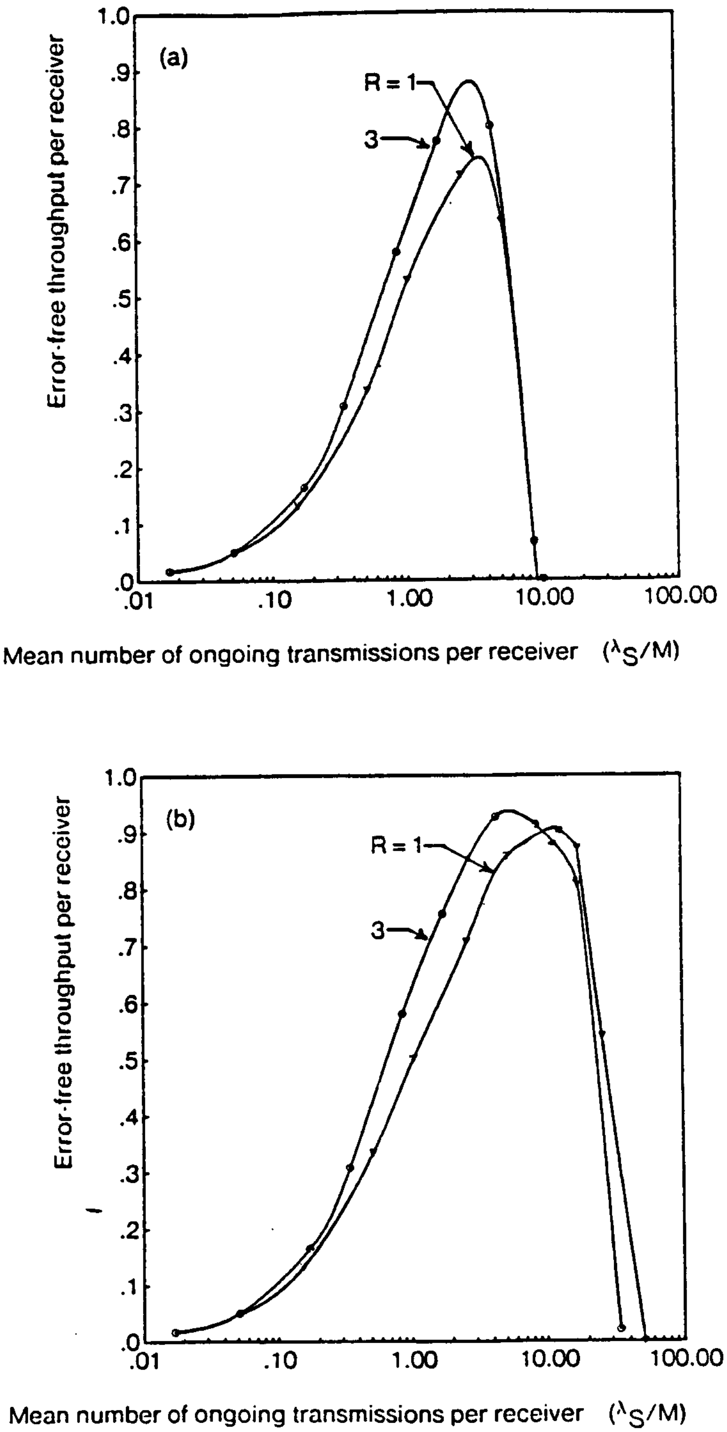


Fig. 4.22 Error-free throughput with FCAA. a) 256 chips per bit; b) 1024 chips per bit. $M = 3$, $\frac{1}{\mu} = 50$.

begins to fall off sharply.

4.5.6 Permitting S to Transmit

At the outset, it should be noted that a supernode is less likely than a conventional node to be subject to the half-duplex constraint. For example, a single antenna can serve all receivers (and no transmitters), thus constituting little overhead. Furthermore, in situations such as the terrestrial hub of a two-hop satellite network, the hub is always full duplex since the up-link and down-link use nonoverlapping spectral ranges. It should also be noted that the incorporation of the fact that a half-duplex supernode may be transmitting has no effect on the relative inbound throughput of the different architectures and code-assignment policies. Nevertheless, this issue is addressed briefly for the sake of completeness.

Let us consider the following policy for the operation of S: transmission may not commence while any of the receivers are busy; whenever all receivers become idle. S commences transmission immediately if it has packets for transmission; if there are no packets for transmission, it must wait until at least one packet is received. The rule for terminating a transmission epoch is not specified; however, once S stops transmitting, it must wait for a reception epoch before it may transmit again. Some portions of this set of rules are realistic, (e.g. no transmissions may commence when engaged in reception,) while others would be modified slightly in a realistic situation. Note, however, that the transmission policy is consistent with the desire to operate all transmitters together, which was explained in section 3.

From the above set of rules, it follows that S alternates between reception and transmission epochs. A reception epoch begins upon termination of transmissions. Initially, all receivers are idle; then, some packets are received. (This idle-busy cycle may be repeated any number of times, and is considered a single reception epoch.) The beginning of a transmission epoch always coincides with the last busy receiver becoming idle, and (obviously) terminates with all receivers idle. Note that idle times (no transmissions or receptions) are considered to be part of a reception epoch.

The calculation of inbound throughput for a nontransmitting supernode were based on cycles that began and ended with the arrival of a receivable packet to an empty system. Comparing this with the reception epoch that has just been defined, one observes that the latter consists of several such cycles, but is missing the time interval from the instant that the last receiver becomes idle in the last cycle of the epoch until the arrival time of the next receivable packet. On the other hand, the reception epoch contains an extra time interval at the beginning, namely the time from end of transmissions until the arrival of the first receivable packet. Due to the memoryless nature of the arrival process, and the meaningless difference in side-information provided by the two states, these two intervals cancel out. Therefore, the analysis that was presented for a nontransmitting supernode remains valid, and the new throughput can be obtained by multiplying those results by the fraction of time in which S is not transmitting. This fraction can be estimated in each specific case, but is beyond the scope of this discussion.

4.6 Full-Duplex Nodes

There are cases in which a node can transmit and receive concurrently. One example is a local-area network which uses low-attenuation broadcast media, so that the levels of received and transmitted signals are similar and the interference with a node's reception due to its own transmission is similar to the interference caused by a transmission of some other node. Another example would be a packet radio network in which the transmitter and the receiver of a node are not collocated, or at least use different antennas. In this section, the results of the previous sections are adapted to the case of full-duplex nodes.

4.6.1 Multiple Receivers and Transmitters

The results obtained for the unslotted model remain valid, since they were derived under the assumption that $\mathcal{S}$ never transmits. For multiple receivers in the slotted case, the expression for inbound throughput (4.7) is simply evaluated for $p_0 = 0$, regardless of the actual value of p_0. The plot of $S_{in}/(1 - p_0)$ in Fig. 4.4 becomes a plot of S_{in}.

With multiple transmitters, time-synchronization is no longer necessary in order to increase the availability of $\mathcal{S}$ for reception. However, we argue that it should still be used in order to maximize $\mathcal{S}$'s outbound throughput, and justify this as follows. Given the environment, characterized by $N, p, Q, P_{ET}(k)$ and $P_S(l)$, along with the number $(T')$of $\mathcal{S}$'s transmitters that are transmitting in a given slot, it is possible to obtain the expected value of $\mathcal{S}$'s outbound throughput in that slot. Let T be the value of T' that maximizes the expected value of this throughput. Clearly, equipping $\mathcal{S}$ with T transmitters and operating them in all time slots will maximize the outbound throughput in all slots. This, however, is exactly time synchronization. Furthermore, T_{opt} is the one that was obtained in section 3.3. Finally, the expression for outbound throughput is the same as (4.12), divided by $(1 - p)$. (If we assumed that only $\mathcal{S}$ is full duplex, (4.12) would remain valid.)

When the desired outbound throughput is smaller than the maximum, it can be achieved either by reducing the number of transmitters or by setting $p_0 < 1$. There is no difference in terms of $\mathcal{S}$'s throughput, but there may be a difference in the efficiency and, consequently, in the throughput of the other nodes. Whenever $P_S(l)$ is concave, the better approach is to leave $p_0 = 1$ and reduce T.

The combination of multiple receivers and transmitters need not be considered, since they operate independently.

4.6.2 Link Masking

If only $\mathcal{S}$ is full duplex, the results obtained for $p_0 = 0$ are valid for all values of p_0. If all nodes are full duplex, one can construct a unary tree of all nodes, i.e., a chain, such that each node can receive traffic for $\mathcal{S}$ from only one of its neighbors and can transmit such traffic only to one of its neighbors. If there is no traffic other than $\mathcal{S}$'s inbound traffic, this can result in an inbound throughput of 1.0. However, this is highly impractical for several reasons: assuming that a packet originates from any of $\mathcal{S}$'s neighbors with equal probability, a packet has to travel $N/2$ hops on average, which may cause delay to be prohibitive; this also amounts to a very low efficiency of channel usage. The mean number of ongoing transmissions is $N/2$, which may even exceed the capacity of the channel, causing a throughput of 1.0 not to be achievable.

Realizing that the chain idea is impractical, the next step is for $\mathcal{S}$ to authorize two of its neighbors to transmit to it. Since those neighbors are also full duplex, their inbound throughputs can be $1/e$, which is more than half of $\mathcal{S}$'s maximum inbound throughput (0.5). Consequently, we are back to 2-hop link masking, with the difference that now an inbound throughput of 0.5 can be achieved for all values of p_0. Transmission rates with direct transmissions and for the 2nd hop of 2-hop link masking can be obtained from the appropriate equations in section 4 by setting $p_0 = 0$. The rate for the 1st hop of 2-hop link masking can be obtained from the appropriate equation by setting $p_1 = 0$.

4.7 Summary

Equipping a node with several receivers and transmitters increases its inbound and outbound throughput, respectively. The increase is eventually limited by channel capacity. The optimal number of receivers is infinite, although there is little to be gained beyond a certain number. The optimal number of transmitters is finite. Since a node cannot receive while transmitting, it is important to enforce time-synchronization between the node's transmitters. Code-synchronization must also be used. This applies to slotted as well as unslotted systems.

In an unslotted system, time-capture permits various ways of assigning codes to receivers, and permits the M-receiver supernode to have a higher inbound throughput than M independent single-receiver nodes. With fixed assignment, the optimal number of receivers that should share a common code is higher for long packets than for short ones, increases with a decrease in channel capacity or an increase in the level of background traffic, and decreases with an increase in the level of inbound traffic. We note that the case in which each of the M receivers has its own code is the same as M separate nodes. Although maximum throughput can be quite insensitive to the number of receivers that share a common code, a significant improvement in efficiency can be achieved by the proper selection. Dynamic reassignment of codes to receivers improves throughput as well as efficiency.

In situations wherein only long-term uniformity of code usage can be assumed, the advantage of DCAA over FCAA is much more pronounced than suggested by our results, since DCAA would adapt to the transient skew in code usage, whereas FCAA would not. In the extreme case that all packets are arriving with a single code, the throughput advantage of DCAA-RA over FCAA would be on the order of $(M - N_c) : \frac{M}{N_c}$, and that of DCAA-OA over FCAA would be on the order of $N_c : 1$.

It is interesting to observe that, due to time capture, the throughput of an individual receiver with the pure ALOHA access scheme can approach 1 (for long packets), whereas that with slotted ALOHA is limited to $\frac{1}{e}$. The best results can be achieved with a minislotted scheme, in which all transmissions begin at the beginning of a slot and the slot lengths equals the preamble length.

For a slotted system, masking all but two of a node's incoming links can increase its inbound throughput by up to 36%; furthermore, at high throughput levels, particularly when S itself transmits frequently, link masking is more efficient than direct transmissions. Unlike multiple receivers and multiple transmitters, link masking requires no additional hardware, since it makes use of otherwise lightly utilized hardware in neighboring nodes. The protocol required to support link masking is very simple and robust. In unslotted systems, link masking has limited application due to time capture.

The number of supernodes that can coexist (efficiently) in the same region of a network is limited primarily by channel capacity; the availability of codes could also be a limiting factor, but more often than not this is not the case [59].

Delay was not addressed directly in this discussion. Nevertheless, whenever the throughput of one architecture exceeds that of another for all arrival rates, that architecture is also superior in terms of delay.

Chapter 5

Conclusions and Future Research

5.1 Conclusions

Single-hop topologies and communication networks that employ shared media are usually considered synonymous with a single broadcast bus, or at most several replicas of such a bus. In chapters 2 and 3, we set out to show that, whenever stations are equipped with multiple transmitters and receivers, this need not and perhaps often should not be the case.

In chapter 2, the design space of single-hop interconnections among stations with multiple transmitters and receivers was shown to be quite rich. Selective-broadcast interconnections were defined and classified, and their performance was then studied. The basic idea in these interconnections is that achieving single-hop connectivity between two such stations does not require that they be able to communicate with each other using any of their transmitters and receivers. This permits the construction of a large number of subnetworks, each of which interconnects only a subset of the stations.

The richness of the design space was shown to be greatly enhanced if unidirectional media are employed. The added richness stems from the fact that, with unidirectional media, the sets of receivers that hear two transmitters need not be identical or disjoint, as is the case with bidirectional media. To facilitate the study of these interconnections, a new criterion for successful concurrent communication over two paths was proposed, and its properties were established.

For a uniform traffic pattern, as well as many others, certain simple $\mathcal{SBI}$'s were shown to permit a level of concurrency which grows quadratically with C, the number of transmitters and receivers per station. The performance for other traffic patterns can be better or worse, depending on whether or not the $\mathcal{SBI}$ is tailored to the specific pattern. Lastly, $\mathcal{SBI}$'s were shown to apply to a variety of domains.

In chapter 3, issues pertaining to the implementation of $\mathcal{SBI}$'s for communication were addressed. A fiber optic implementation of an $\mathcal{SBI}$ that permits a level of concurrency which grows according to C^2 was shown to be optimal in terms of power budget. Also, the cost of passive interconnection components for such an $\mathcal{SBI}$ need not be higher than that of C replicas of the broadcast bus. Although most $\mathcal{SBI}$'s require certain modification in order to be operated with existing channel-sensing access schemes, the use of such schemes nevertheless increases the performance advantage of $\mathcal{SBI}$ over a straightforward C-fold replication of the single bus.

$\mathcal{SBI}$'s permit the total communication bandwidth of a single user to be C times lower than the capacity of the network. Furthermore, the utilization of the transmitters and receivers can be C times higher than with C replicas of a broadcast bus. This is of particular importance to VLSI implementations of stations, (e.g. for memory-processor interconnections,) in which pin count is often a limiting factor.

In summary, it was shown that equipping stations with multiple transmitters and receivers permits the construction of interconnections which can greatly improve the ratio of cost to performance relative to that with a single bus.

In networks which use a single broadcast bus, nodes have always been equipped with a single transceiver. We showed that in the case of spread-spectrum with a nonuniform traffic pattern, equipping busy nodes with several transmitters and receivers is crucial in obtaining reasonable performance. This holds regardless of the variant of spread-spectrum that is being used. In chapter 4, we explored the design and performance of such nodes. We showed that their transmitters must all be operated together, that the optimal number of transmitters is finite, and that an appropriate assignment of spread-spectrum codes to the receivers can significantly increase the inbound throughput of the node relative to that of the same number of receivers residing in different nodes.

To conclude, we have shown that the performance of communication networks that employ shared media can indeed be greatly enhanced beyond the obvious if stations are equipped with multiple transmitters and receivers.

5.2 Suggestions for Additional Research

Selective-broadcast interconnections were discussed only in the context of narrowband channels. However, they can clearly be used with spread-spectrum. While in some cases, such as $\mathcal{SBI}$'s with disjoint subnetworks, this extension is trivial, there may be less trivial results when $\mathcal{SBI}$'s with overlapping subnetworks are considered, particularly if they are not equal-degree $\mathcal{SBI}$'s. It is also worth noting that CDMA, being a non-sensing access scheme, is particularly suitable to unidirectional $\mathcal{SBI}$'s.

The study of $\mathcal{SBI}$'s focused on the topology, and the discussion of access schemes emphasized the adaptation of $\mathcal{SBI}$'s to operate with single-bus access schemes.

However, as was pointed out in chapter 3, special access schemes can be devised for the combined operation of the entire $\mathcal{SBI}$. This would be particularly interesting for multi-path $\mathcal{SBI}$'s, such as the hybrid $\mathcal{SBI}$-$\mathcal{PBI}$.

We have shown that equal-degree $\mathcal{SBI}$'s which can simulate high-flux multi-hop interconnections can be constructed. These and others may be useful in the execution of various distributed functions. For example, sorting entries that reside in different stations. It would be interesting to look into the suitability of $\mathcal{SBI}$ for the efficient execution of such algorithms.

We have constructed an equal-degree $\mathcal{SBI}$ and a schedule, which can achieve an average concurrency of $1.81 \cdot C^2$ for a uniform traffic pattern. However, the upper bound on the concurrency of $\mathcal{SBI}$'s for uniform traffic patterns and single-hop communication is an open problem.

In our study of supernodes in spread-spectrum networks, we have not exhausted the design flexibility offered by CDMA and time-capture. For example, one could construct a ***super-link*** between two busy nodes that communicate extensively with each other. Also, more detailed performance analysis, using more accurate models and perhaps addressing delay in addition to throughput, may be of interest, although the basic results of this work are expected to hold.

Appendix A

Optimality of the Wavelength-Assignment Algorithm for the SDM-WDM Implementation of an $\mathcal{SBI}$

In this appendix, we present a proof of the optimality of the wavelength-assignment algorithm used in the hybrid SDM-WDM implementation of the single-path, unidirectional, equal-degree $\mathcal{SBI}$ with disjoint subnetworks and grouping.

Wavelength-assignment algorithm. Arrange the couplers in bunches of G, where G is the least common multiple of C_T and W; number the bunches consecutively modulo W, beginning with 0. Next, number the couplers within each bunch consecutively modulo W, beginning with the bunch's number. The numbers correspond to distinct wavelengths.

At the outset, recall that in all cases, it is assumed that W divides $C_T \cdot C_R$. The optimality proof proceeds by initially presenting and proving statements that are true in several cases, and then using them to prove optimality for each case. Proofs for cases (4) and (5) are not presented, since they follow directly from those for (2) and (3).

After numbering the couplers, circle every C_Tth coupler, beginning with the first. (The C_R circled couplers are the ones to which the receivers of the first group

$$\left.\begin{array}{ccccccccc}\textcircled{0} & 1 & 2 & 3 & 4 & 5 & \textcircled{6} & 7 & 8 \\ 0 & 1 & 2 & \textcircled{3} & 4 & 5 & 6 & 7 & 8\end{array}\right\}\text{Bunch \#1}$$

$$\left.\begin{array}{ccccccccc}\textcircled{1} & 2 & 3 & 4 & 5 & 6 & \textcircled{7} & 8 & 0 \\ 1 & 2 & 3 & \textcircled{4} & 5 & 6 & 7 & 8 & 0\end{array}\right\}\text{Bunch \#2}$$

Fig. A.1 Numbering, circling and bunching of couplers.

of RS's connect.) Next, arrange the couplers in rows of W, such that the first W couplers are in the first row, the next W couplers are in the 2nd row, etc.; there are $C_T \cdot C_R/W$ such rows.

Let $k \triangleq \frac{G}{W}$; i.e., k is the smallest positive integer such that C_T divides $k \cdot W$. Clearly, $k \leq C_T \cdot C_R/W$, since both W and C_T divide $C_T \cdot C_R$. Next, arrange the rows in bunches of k, such that the first k rows are in the first bunch, the k following rows are in the next bunch, etc. (Note that the bunching of the rows coincides with the bunching of the couplers in the assignment algorithm.) Fig. A.1 illustrates the numbering, circling and bunching for $C_T = 3$, $C_R = 4$, $W = 6$. Since there are exactly G couplers per bunch, and W couplers per row, it follows that, within any given bunch, all couplers in any given column are assigned the same number.

Lemma 1. The first coupler in each bunch is circled, and there are no two rows within a bunch that have a circled coupler in the same position; furthermore, the positions of all circled couplers are identical for all bunches.

Proof. Follows immediately from the fact that $k \cdot W$ is the smallest common period of the coupler-circling process (period C_T) and the rows (period W).

Lemma 2. All the circled numbers in any given bunch are equispaced modulo W.

Proof. By contradiction; if the numbers are not equispaced, there must be a set of three numbers $\{w_1, w_2, w_3\}$ (where $w_i \in 0..(W-1)$) such that (i) they are are all circled in a certain bunch, (ii) no other circled numbers are encountered when counting (modulo W) from w_1 through w_2 to w_3 and (iii) $(w_2 - w_1) \bmod W \neq (w_3 - w_2) \bmod W$. Let us assume that the smaller of the two spacings is between w_1 and w_2. From the circling procedure and the aforementioned common periodicity, it follows that $w_2 = (w_1 + k' \cdot C_T) \bmod W$ for some positive integer k'. However, it also follows that $w_3' \triangleq (w_2 + k' \cdot C_T) \bmod W$ is circled and that $(w_3' - w_2) \bmod W = (w_2 - w_1) \bmod W$. This, in turn, means that a circled number is encountered in the process of counting modulo W from w_2 to w_3, which contradicts the assumptions.

Proposition A.1. Whenever $W \geq C_R$ and W divides $C_T \cdot C_R$, any station in the first RS group is assigned C_R different wavelengths.

Proof. Since each bunch consumes $k \cdot W/C_T$ *different* circled *numbers* (lemma 1), and those are equispaced (lemma 2), the minimum difference between any two circled numbers within a bunch is C_T/k. Consequently, there can be up to C_T/k bunches of $k \cdot W$ couplers, such that the first coupler in the first bunch is numbered 0, the first coupler in the 2nd bunch is numbered 1, etc., (and the remaining couplers in each bunch are numbered accordingly,) such that no two circled couplers have the same number. The actual number of bunches is $(C_T \cdot C_R)/(k \cdot W)$ and, since $W \geq C_R$, this is less than or equal to C_T/k. In other words, no two circled couplers in the entire $\mathcal{SBI}$ have the same number, and each receiver of RS's in the first group is thus assigned a unique wavelength.

Corollary. For any RS, each receiver is assigned a unique wavelength.

Proof. From the circling algorithm, it follows that the last $C_T - 1$ couplers in each bunch are not circled. Also, stations in the 2nd RS group are connected to couplers immediately following the circled ones, stations in the 3rd groups to couplers immediately following those, etc. Therefore, and since there are C_T RS groups, the set of wavelengths used by the ith TS group is obtained by adding $(i-1) \bmod W$ to each of the wavelength numbers used by the first group. Proposition A.1 implies that those are C_R different wavelengths.

Corollary. Each RS makes equal usage of all wavelengths that are assigned to it. (It uses each wavelength exactly once.)

Proposition A.2. Whenever $W > C_T$ and W divides $C_T \cdot C_R$, each TS is assigned C_T different wavelengths.

Proof. The transmitters of any given station are connected to C_T consecutive couplers. Also, consecutive couplers are assigned consecutive numbers (mod W), with the exception that the number is incremented at each crossing of a bunch boundary. Since there are at least W couplers in each bunch and $W > C_T$, it follows that the numbers assigned to C_T consecutive couplers cover a range of at most $(C_T + 1)$ consecutive values, which is less than or equal to W; consequently, all those numbers are different.

Proposition A.3. The above also holds when $W = C_T$.

Proof. In this case, $G = W = C_T$, and bunch boundaries coincide with the boundaries between the couplers used by different TS groups. Therefore, it is obvious that the C_T transmitters of any given station use $C_T = W$ different wavelengths.

Corollary. Whenever $W \geq C_T$ and W divides $C_T \cdot C_R$, each TS makes equal usage of all wavelengths that are assigned to it.

Propositions A.1, and A.2 constitute the optimality proof for case (1). A.1 and A.3 prove optimality for case (2).

Proposition A.4. The assignment algorithm is also optimal when W divides C_T.

Proof. Since bunch boundaries coincide with TS-group boundaries, it follows that each TS uses all W wavelengths; furthermore, each wavelength is assigned to exactly C_T/W of its transmitters. Since the first couplers in each bunch are assigned consecutive numbers (mod W), and since each bunch consists of exactly C_T couplers, it immediately follows that receivers of a station in the first RS group are assigned couplers with consecutive numbers. This constitutes a uniform round robin assignment of wavelengths to receivers. Using the same arguments as in the first corollary to proposition A.1, it follows that the same applies to all RS's. This completes the proof for case (3).

References

[1] M. Schwartz, *Computer-Communication Network Design and Analysis*, Prentice-Hall, 1977.

[2] A.S. Tanenbaum,*Computer Networks*, Prentice-Hall, 1982.

[3] *Computer Networks Architectures and Protocols*, P.E. Green, Ed., Plenum Press, 1982.

[4] W. Stallings, *Local networks – An Introduction*, Macmillan, 1984.

[5] *Satellites, Packets, and Distributed Telecommunications – a Compendium of Source Materials*, R.D. Rosner, Ed., Lifetime learning Publications, Belmont CA 1984.

[6] M. Schwartz, *Telecommunication Networks – Protocols, Modeling and Analysis*, Addison Wesley, 1986.

[7] J.M. Mcquillan and D.C. Walden, "The ARPA network design decisions," *Comput. Networks*, vol. 1, pp. 243-289, Aug. 1977.

[8] R.M. Metcalfe and D.R. Boggs, "Ethernet: Distributed packet switching for local computer networks," *Communications of the ACM*, vol. 19, no. 7, pp. 395-403, 1976.

[9] *Local Network Equipment*, H.A. Freeman and K.J. Thurber, Eds., IEEE Comp. Soc. Press, 1985.

[10] *IEEE Commun. Mag.*, vol. 22, no. 1, Jan. 1984.

[11] *IEEE Commun. Mag.*, vol. 24, no. 3, Mar. 1986.

[12] H.J. Siegel, *Interconnection Networks for Large-Scale Parallel Processing*, Lexington books, Lexington, MA, 1985.

[13] C. Clos, "A study of non-blocking switching networks," *Bell System Technical Journal*, vol. 32, no. 2, pp. 406-424, March 1953.

[14] D.H. Lawrie, "Access and Alignment of Data in an Array Processor," *IEEE Trans. Computers*, vol. C-24, pp. 1145-1155, Dec. 1975.

[15] L.R. Goke and G.J. Lipovski, "Banyan Networks for Partitioning Multiprocessor Systems," *First Annual Int'l Symp. Computer Architecture*, Dec. 1973, pp. 21-28.

[16] V.E. Benés, "Optimal rearrangeable multistage connecting networks," *Bell systems technical journal*, vol. 43, no. 4, part 2, pp. 1641-1656, July 1964.

[17] A.B. Strowger, "Automatic telephone exchange," *U.S. patent* no. 447918, March 1891.

[18] D.J. Farber and K.C. Larson, "The system architecture of the distributed computer system – the communications system," *Symp. on Computer Networks*, Polytechnic Institute of Brooklyn, Apr. 1982.

[19] N.F. Maxemchuk, "The Manhattan street network," *Globecom '85*, New Orleans, LA.

[20] M.C. Pease, III, "The indirect binary n-cube microprocessor array," *IEEE Trans. Comput.*, vol. C-26, no. 5, May 1977.

[21] F.P. Preparata and J.E. Vuillemin, "The cube-connected cycles: a versatile network for parallel computation," *Proc. 20th annual IEEE Symposium on Foundations of Computer Science*, pp. 140-147, 1979.

[22] H.S. Stone, "Parallel processing with the perfect shuffle," *IEEE Trans. Comp.*, vol. C-20, no. 2, pp. 153-161, 1971.

[23] C.L. Seitz, "Concurrent VLSI architectures," *IEEE Trans. Comp.*, vol. C-33, no. 12, Dec. 1984.

[24] Special issue on spread-spectrum communications, *IEEE Trans. Commun.*, vol. COM-30, pp. 817–1072, May 1982.

[25] J.Y. Hui, "Pattern code modulation and optical decoding – a novel code-division multiplexing technique for multifiber networks," *IEEE JSAC*, vol. 3, no. 6, pp. 916–927, Dec. 1985.

[26] F.A. Tobagi, "Multiaccess protocols in packet communication systems," *IEEE Trans. Commun.*, vol. COM-28, Apr. 1980.

[27] M.A. Marsan and D. Roffinella, "Multichannel local area network protocols", it IEEE JSAC, vol. 1, no. 5, pp. 885-897, Nov. 1983.

[28] T. Lang, M. Valero and M.A. Fiol, "Reduction of Connections for Multibus Organization," *IEEE Trans. Comp.*, vol. C-32, no. 8, pp. 707-715, Aug. 1983.

[29] M.R. Garey and D.S. Johnson, *Computers and Intractability – a Guide to the Theory of NP-Completeness*, W.H. Freeman and company, 1979.

[30] L.G. Valiant and G.J. Brebner, "Universal schemes for parallel computation," *Proc. 13th Annual ACM Symposium on the Theory of Computing*, pp. 263-277, 1981.

[31] S. Kirkpatrick, C.D. Gelatt, Jr., and M.P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, May 1983.

[32] A. El Gamal and I. Sperling, " Good codes via simulated annealing," *Proc. Conf. on Statistical Mechanics Applications in Engineering and Biology*, IBM T.J. Watson Research Center, Yorktown Heights NY, Apr. '84.

[33] L. Kleinrock, *Queueing Systems, vol. I: Theory.* Wiley-Interscience, New York, NY, 1975.

[34] R.C. Stearns, C.K. Asawa and S.K. Yao, "Angular division multiplexer for fiber communication using graded-index rod lenses," *IEEE J. of Lightwave Technology* vol. 2, p. 358. 1984.

[35] P.P. Deimel et al, in *Digest of Topical Meeting on Optical Fiber Communication* (Optical Society of America, Washington, D.C.), paper TUC4, p. 32. 1985.

[36] M.E. Marhic, *Opt. Lett.*, vol. 9, p. 368. 1984.

[37] R.W. Boyd, *Radiometry and the Detection of Optical Radiation*, John Wiley & sons, New York, 1983.

[38] J.W. Goodman, "Fan-in and fan-out with optical interconnections," *Optica Acta*, vol. 32, no. 12, pp. 1489-1496, 1985.

[39] T.V. Muoi, "Receiver design for high-speed optical-fiber systems," *IEEE J. of Lightwave Technology*, vol. 2, pp. 243-267, June 1984.

[40] In *Lasers and Applications*, vol. 4, p. 18. January 1985.

[41] N.A. Olsson, J. Hegarty, R.A. Logan, L.F. Johnson, K.L. Walker and L.G. Cohen, in *Digest of Topical Meeting on Optical Fiber Communication* (Optical Society of America, Washington, D.C.), paper WB6, p. 88. 1985.

[42] N. Abramson, "The ALOHA system – Another alternative for computer communications," *AFIPS Conference Proceedings, 1970 Fall Joint Computer Conference*, vol. 37, pp. 281–285.

[43] L. Kleinrock and F.A. Tobagi, "Packet switching in radio channels: part I: carrier-sense multiple-access modes and their throughput-delay characteristics," *IEEE Trans. Commun.*, vol. COM-23, pp. 1400-1416, Dec. 1975.

[44] F.A. Tobagi and V.B. Hunt, "Performance analysis of carrier sense multiple access with collision detection," in *Proc. Local Area Commun. Network Symp.*, Boston, MA, May 1979.

[45] F. Tobagi, F. Borgonovo and L. Fratta, "Express-net: a high-performance integrated services local area network," *IEEE Journal on Selected Areas in Communications*, vol. SAC-1, Nov. 1983.

[46] M. Fine and F. Tobagi, "Demand assignment multiple access schemes in broadcast bus local area networks," *IEEE Trans. Computers*, vol. C-33, no. 12, pp. 1130-1159, Dec. 1984.

[47] J.O. Limb and C. Flores, "Description of Fasnet, a unidirectional local area communications network," *Bell System Technical Journal*, Sep. 1982.

[48] M.B. Pursley, D.V. Sarwate and W.E. Stark, "Error probability for direct-sequence spread-spectrum multiple-access communications — Part I: upper and lower bounds," *IEEE Trans. Commun.*, vol. COM-30, pp. 975–985, May 1982.

[49] W.E. Stark, "Coding for frequency-hopped spread-spectrum communication with partial-band interference — Parts I, II," *IEEE Trans. Commun.*, vol. COM-33, pp. 1036–1057, Oct. 1985.

[50] J. Storey and F. Tobagi, "Throughput performance of a direct sequence CDMA packet radio network," *Stanford University Computer Systems Laboratory Report*, SEL TR 85-277, June 1985.

[51] E. Parker, "Micro Earth Stations as Personal Computer Accessories," *IEEE Proc.*, vol. 72, no. 11, pp. 1526-1531, Nov. 1984.

[52] R. Kahn, S. Gronemeyer, J. Burchfiel and R. Kunzelman, "Advances in packet radio technology," *IEEE Proc.*, vol. 66, pp. 1468–1496, Nov. 1978.

[53] L.G. Roberts, "ALOHA packets with and without slots and capture," *Comput. Commun. Review*, vol. 5, pp. 28-42, Apr. 1975.

[54] G. Clark, Jr. and J. Cain, *Error-Correction Coding for Digital Communications*, Plenum Press, New York, 1981.

[55] I. Gitman, "On the capacity of slotted ALOHA networks and some design problems," *IEEE Trans. Commun.*, vol. COM-23, pp. 305–317, Mar. 1975.

[56] F.A. Tobagi, "Performance analysis of packet radio communication systems," *Proceedings of the IEEE Nat. Telecom. Conf.*, Dec. 1977, pp. 12:6-1 – 12:6-7.

[57] G. Akavia and L. Kleinrock, "Performance tradeoffs and hierarchical designs of distributed packet-switching communication networks," *UCLA technical report* no. UCLA-ENG-7952, Sep. 1979.

[58] H. Takagi and L. Kleinrock, "Output processes in contention packet broadcasting systems," *IEEE Trans. Commun.*, vol. 33, no. 11, pp. 1191–1199. Nov. 1985.

[59] Personal correspondence with John H. Cafarella, MICRILOR, P.O.Box 624, Swampscott, MA 01907

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER 87-321	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) CONCURRENT COMMUNICATION AMONG MULTI-TRANSCEIVER STATIONS OVER SHARED MEDIA		5. TYPE OF REPORT & PERIOD COVERED TECHNICAL REPORT
		6. PERFORMING ORG. REPORT NUMBER 87-321
7. AUTHOR(s) Yitzhak Birk		8. CONTRACT OR GRANT NUMBER(s) NASA NAG2-292 NASA NAGW-419
9. PERFORMING ORGANIZATION NAME AND ADDRESS Stanford Electronics Laboratories Stanford University Stanford, CA 94305-2192		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS National Aeronautics and Space Administration Washington, D.C. 20546		12. REPORT DATE March 1987
		13. NUMBER OF PAGES 223
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Resident Representative Office of Naval Research Durand 165 Stanford University, Stanford, CA 94305-2192		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Presently, most local-area networks employ a single broadcast bus to interconnect single-transceiver stations. In order to increase a network's throughput beyond a single bus's data rate without using dedicated switching nodes, multiple buses and multi-transceiver stations are required. We explore the design space of single-hop interconnections among such stations; i.e., interconnections that provide a passive transmission path between any two stations. For example, we present interconnections whose throughput can grow *quadrati-*

cally with the number of transmitters and receivers per station. They consist of a collection of buses, each of which interconnects only a proper subset of the stations using one of their transceivers. Yet, for any two stations, there is at least one bus to which they are both connected. We refer to these as selective-broadcast interconnections, or $\mathcal{SBI}$'s. The use of unidirectional media significantly enriches the design space of $\mathcal{SBI}$'s since, unlike with bidirectional media, the sets of receivers that hear two transmissions need not be identical or disjoint. A graph-theoretic criterion for determining whether or not transmissions over a specified pair of paths would interfere with each other is established. It is then used in studying the performance of various $\mathcal{SBI}$'s. Implementation-related issues, such as power budget in fiber optic implementations, are discussed in the context of local-area networks. Lastly, the concept of $\mathcal{SBI}$'s is shown to also apply to memory-processor interconnection, as well as to additional domains.

A spread-spectrum channel can accommodate several concurrent successful transmissions, and a single-transceiver node can thus utilize only a small fraction of the channel's capacity. In order to allocate the appropriate fraction of capacity to a "busy" node, we propose to equip it with several transmitters and receivers, thereby turning it into a "supernode". Several architectures and operation policies for supernodes are suggested and compared; it is shown that a supernode can significantly outperform a collection of independent conventional nodes with the same total numbers of transmitters and receivers. Packet-radio networks with half-duplex nodes, as well as networks with full-duplex nodes, are considered.